



# **RSI-ECAT-Master**

## **User's manual**




Version 2012/09/21

**Micronet**

Micronet Company  
3-8-11 Horiwari, Kamisu-shi, Ibaraki  
314-0135, Japan

Tel : 0299-90-1733  
Fax : 0299-92-8557  
URL : <http://www.mnc.co.jp/>

【About the notation in this manual】

	Hint: The information for deepening your understanding of the product is described.
	Attention: May result in damage to data or hardware
	Remark: We describe the important information about the product.

- ☐ Reproducing the contents of this manual and reprinting all or a part of software or attachments without notice is forbidden.
- ☐ The contents of this product might be changed without any advance notice.
- ☐ If you are aware of some doubtful contents of this product or omissions in this manual, sorry to trouble you, but please contact our company.
- ☐ EtherCAT® is a registered trademark of Beckhoff Automation GmbH, Germany and a patented technology.
- ☐ Microsoft is a registered trademark of Microsoft Corporation, USA.
- ☐ Windows® 7 and Windows® XP are registered trademarks of Microsoft Corporation, USA.
- ☐ TenAsys is a registered trademark of TenAsys Corporation, USA.
- ☐ INtime and iRMX are registered trademarks of TenAsys Corporation, USA.
- ☐ Koenig, KPA and Koenig Prozessautomatisierungs GmbH are registered trademarks of Koenig Prozessautomatisierungs GmbH, Germany.
- ☐ Beckhoff GmbH and Beckhoff are registered trademarks of Beckhoff GmbH, Germany.

## 1 Index

1	Index .....	3
2	Introduction .....	5
3	About this product .....	5
4	The main function of this product.....	5
5	Term Description .....	8
6	Functional Constitution Figure.....	12
7	Connection of Slave Apparatus.....	13
8	Connection in Redundancy.....	13
9	Requirements for Operation.....	13
10	RSI-ECAT-Master Configuration File.....	13
10.1	EtherCAT Protocol Driver RSL .....	13
10.2	INtime process RTA to perform the handshake with EtherCAT protocol .....	13
10.3	EtherCAT slave various drivers RSL.....	13
10.4	API-RSL.....	13
11	RSI-ECAT-Master configuration.....	13
12	Start procedure flow.....	13
13	Indispensable setting items.....	13
13.1	Create ENI(EtherCAT Network Information) File .....	13
13.2	An INtime memory is set as 64 MB or more. ....	13
13.3	Specification of Start Node.....	13
13.4	Specification of NIC(Network Interface Controller) to be used .....	13
13.5	Mandatory alias.....	13
13.6	Station Alias = Alias.....	13
14	Pass of Network Device.....	13
15	The ENI file creation procedure by RSI-ECAT-Studio.....	13
15.1	The things to prepare.....	13
15.2	Pre-Preparation.....	13
15.3	Application of ESI File.....	13
15.4	Creation of ENI File.....	13
15.5	The ENI file created to RSI-ECAT-Master is arranged or updated. ....	13
16	Alias setup steps by RSI-ECAT-Studio.....	13
16.1	The things to prepare.....	13
16.2	Pre-Preparation.....	13

16.3	Detection of the EtherCAT slave connected .....	13
16.4	Setup of temporary alias .....	13
16.5	Applied as Regular alias .....	13
16.6	Cautions of alias setup .....	13
17	How to start the RSI-ECAT-Master .....	13
18	How to autostart RSI-ECAT-Master .....	13
19	How to Exit the RSI-ECAT-Master .....	13
20	Starting parameter .....	13
20.1	NIC selection parameters main line .....	13
20.2	NIC selection parameters redundancy line .....	13
20.3	Cyclic time specification parameter .....	13
20.4	Thread base priority change parameter .....	13
20.5	Starting delay parameter .....	13
20.6	Output parameters when error occurs .....	13
21	API .....	13
21.1	EH API .....	13
21.2	Slave API .....	13
22	File composition .....	13
23	The concept of a node .....	13
24	The check of a starting state .....	13
24.1	Call the API .....	13
24.2	Check by INtime Explorer .....	13
25	Master default configuration .....	13
26	ecatmkpa.ini setting item .....	13
27	Sample Code .....	13
28	Sample Program .....	13
28.1	How to run the sample program .....	13
28.2	Checking of Operations of Sample Program .....	13
28.3	Edit of sample program .....	13
29	Troubleshooting .....	13
29.1	The view of an error message table .....	13
29.2	Error Message Table .....	13
30	About License .....	13

## 2 Introduction

Thank you for your purchasing our products.

The purpose of this document is to describe how to handle the product RSI-ECAT-Master in general. For detailed reference on application interface functions (API) library that is included in this product, please refer to the help file that is introduced into the computer by the installation of this product.

## 3 About this product

Without installing additional hardware such as interface expansion board to industrial computers, this RSI-ECAT-Master product is a software which provides communication can realize EtherCAT<sup>®</sup> protocol via the network interface port as standard equipment. And it is one of the options drivers for real-time OS INtime. By using the EtherCAT<sup>®</sup> protocol, it is possible to access EtherCAT<sup>®</sup> slave devices by various vendors.

The network composition information: ENI file (EtherCAT Network Information) which can be created with optional RSI-ECAT-Studio (option) is needed for operation of this product.

Although this book will explain the handling of the product RSI-ECAT-Master, on the product configuration, Please understand that the handling of the product “RSI-ECAT-Studio (option)” is also mentioned.

## 4 The main function of this product

### **EtherCAT master**

It is a main station (master) which manages the EtherCAT slaves by various vendors as an EtherCAT protocol. This product provides EtherCAT master functions only by software and network interface controller which current standard PCs have as a standard.

### **Many NIC compliance**

Compatible to almost all of network interface controllers that current standard PCs have such as Intel chip set so that you can use it to communicate EtherCAT control.

### **Real-time guarantee**

RSI-ECAT-Master can run on INtime software so that we can give your guarantee for the time of processing operation.

### **RSI-ECAT-Master API**

Interface libraries (API) which control an EtherCAT master / slave from INtime applications is provided.

### **Online help**

An API reference is offered by the help file of CHM form so that INtime application development can be performed efficiently.

### **Cyclic operation**

By high-speed periodic communication, the process image (PI) with all connected slaves is updated automatically.  
Setting values of a cyclic can be changed.

### **Virtual-I/O space (VIOS)**

Cyclic operation updates automatically in the process image (PI) of all the slaves arranged in consecutive memory spaces. Two spaces such as Input VIOS and Output VIOS exist and each has the address "0".

You treat it as 16 bit [ 8 bits/]/32bit access by setting the address as a parameter.

### **ENI reading function**

To acquire the EtherCAT slave configuration, RSI-ECAT-Master reads (ENI) EtherCAT Network Information file at startup. ENI file is created (option) RSI-ECAT-Studio.

### **Master State control**

The application can control the state of the EtherCAT master communication (Init, Pre-Operational, Safe-Operational, Operational) in the RSI-ECAT-Master API.

### **Alias function**

Alias is the ID number that can be set to individual slaves that are specified in the standard EtherCAT. Alias is always used for a particular slave on RSI-ECAT-Master. The slave must be configured in a number of non-overlapping unique alias. Please refer to the instruction manual of how to set the alias for each slave device of each vendor.

### **Slave search function**

Regarding Alias No. / Vendor ID / Product ID as the key, you can check the existence of the target slave as well as you can get the address information such as virtual I/O space (VIOS) by using the provided API, and even if some slave are unknown, they can be controlled.

### **SDO control function**

The API is provided to read / write object dictionary (OD) space of the slave so that you can change the slave operation dynamically.

### **EtherCAT slave driver RSL (SDR)**

There is library file (RSL) function so that it might be easy to treat a slave specific function, RSI-ECAT-Master is automatically loaded when there is the correspond SDR. SDR mainly supplies slave specific API and processing of a category memory. Even though there is no SDR in a slave, you can control by virtual-I/O space (VIOS).

### **Category memory function**

This is the automatically categorized / consecutive data space which is each of 4 categories such as analog input (AI), analog output (AO), digital input (DI), digital output (DO) from process image (PI). Each category has the consecutive channel (CH) number from "0" and you can operate DI/DO by 16bits and AI/AO by 32bits per 1 channel. Even if you'd like to control many signals with many slaves, you can do the read / write in block as consecutive channels per each signal category without being conscious of each slave.

### Create SDR

The EtherCAT slave driver RSL (SDR) can be created by users themselves according to the structure which RSI-ECAT-Master define. In SDR internal processing, Slave API which RSI-ECAT-Master products offer can be used. Refer to "RSI-ECAT SDR Making Guide" of a separate volume for your reference.

### Sample Programs

The INtime application sample programs by C language which shows the usage of RSI-ECAT-Master API are offered.

### Multi INtime kernel mode

It corresponds to the multi-kernel mode of INtime which can be constituted by the multi-core CPU environment, so that you can use RSI-ECAT-Master and its related INtime application separately for the purpose of a load sharing.

### Processing priority adjustment

You can change settings of thread priorities such as cyclic operation process according to the system requirements.

### An event / event filtering function

Provides the API to capture some events such as "an error that occurred in the EtherCAT line" as an event. You can filter the unwanted events.

### Slave State check function

Provides the API that can diagnose the connection status of the slave device.

### Hold output state when a communication error occurs

When a communication error has occurred, the signal state of a signal output slave can be held. (It's dependent on the capability of a slave.)

### Distribution Clock /DC (Option)

RSI-ECAT-Master/DC edition corresponds to the Distribution Clock (DC) function which is in the EtherCAT specification. Transmission time of the communication line is taken into account, the simultaneity of the signal input/output of the slave group is guaranteed.

### Redundancy function (option)

RSI-ECAT-Master/RED edition corresponds to the redundancy function which is in the EtherCAT specification. You can keep on communicating with another line if lines or slaves are damaged, by connecting in a ring slave with two Ethernet ports. So you can improve the quality and reliability of your system.

### Hot plug (option)

By the function of a RSI-ECAT-Master/RED edition, if the slave is damaged in on-line, you can exchanged it with the same one to restore your system. An alias number is used for the restoration judgment.

### Optional slave driver (option)

In order to have systems development furthered quickly, the slave drivers of typical slave apparatus are manufactured and sold by our company. It is considered that it is easy to handle by API etc.

## 5 Term Description

### **CoE(CANopen over EtherCAT)**

This means the CANopen protocol service on EtherCAT. It is used for a setup of a communications parameter or a setup of a proper parameter to a device.

CoE is used in some APIs such as EhReadOD(), EhWriteOD(), EhReadODByAlias(), and EhWriteODByAlias().

### **DC(Distributed Clocks)**

Synchronous system between EtherCAT master and slave. It adjusts the time of error based on the provided time in order to realize high-precision synchronization. This function is in RSI-ECAT-Master/DC, the RSI-ECAT-Master/RED.

### **EEPROM**

This is a kind of nonvolatile memory and is a ROM which can rewrite the contents. It is often equipped in EtherCAT slaves and you can record information of the each slave and you can read the information. Please use RSI-ECAT-Studio (option) if you'd like to record / read the information of EEPROM.

### **ENI(EtherCAT Network Information)**

This file which contains the settings for configuration of the EtherCAT master device. In order to operate RSI-ECAT-Master, this file is certainly required.

In order to create this file, RSI-ECAT-Studio (option) is needed.

### **ESI(EtherCAT Slave Information)**

It is a file of the XML form that EtherCAT slave definition information is described.

In order to create the slave composition in off-line by RSI-ECAT-Studio, it is certainly needed.

It is usually provided from the slave vendor.

### **EtherCAT(Ethernet Control Automation Technology)**

It is one of the industrial open networks, and is a protocol in which you can do real-time Ethernet communication.

### **NIC(Network Interface Controller)**

It is a network controller for connecting with a network. This document describes it as a generic term of network controllers such as expansion cards, network controllers mounted on board.

### **PI(Process Image)**

It is the composition of the data structure by PDO communication. You can see this composition by RSI-ECAT-Studio (option).

### **VIOS(Virtual I/O Space)**

It is one of the RSI-ECAT-Master original functions.

VIOS is a copy of PDO and you can operate input and output of PDO by operating VIOS.

There are input (IN) and an output (OUT) in VIOS as well as there are input (IN) and an output (OUT) in PDO.

### **Init**

It is a state of the initial communication and is one of ESM. In this state, you cannot use both PDO communication and SDO communication.

### **Index**

In EtherCAT, this mainly means the address of the object defined in an object dictionary.

It is usually written by 4 figures of hexadecimal notations and may include a sub-index.

### **Alias**

It is an another name in order to identify each slave. An alias denotes the numerical value of 1 to 65535 and it is necessary to set it up so that the same numerical value may not exist on a connection.

### **Object Dictionary**

It is based on the object dictionary of CANopen and all the objects are the data structures which consists of an index and a subindex value. It includes data-type object, communication object and application object. You can use APIs such as EhReadOD(), EhWriteOD(), EhReadODByAlias(), EhWriteODByAlias() in order to access object dictionary.

### **Operational**

It is the State in which SDO communication and PDO communication are performed and is one of ESM.

It is the State for the normal operation.

### **Category memory**

It is one of the RSI-ECAT-Master original functions.

It is the automatically categorized / consecutive data space which is each of 4 categories such as analog input (AI), analog output (AO), digital input (DI), digital output (DO) from process image (PI). Each category has the consecutive channel (CH) number from "0" and you can operate DI/DO by 16bits and AI/AO by 32bits per 1 channel. Even if you'd like to control many signals with many slaves, you can do the read / write in block as consecutive channels per each signal category without being conscious of each slave.

### **SDO(Service Data Object)**

It is the service which performs data communication at arbitrary timing using a mail box protocol.

It is mainly used for accessing to object dictionary.

### **Service data communications (SDO communications)**

It is one of the EtherCAT communications and is the communication which performs data communication at arbitrary timing using Service Data Object.

### **Cyclic**

In EtherCAT, it means the event which is repeated regularly and consecutively.

#### **Sub-index**

In EtherCAT, it mainly means the sub-address of the object in an object dictionary.

It is usually written by double figures of hexadecimal number, and the total number of sub-indexes which an index has is stored in sub-index "0".

#### **State, State machine (ESM) (EtherCAT State Machine)**

It is a State machine system of EtherCAT.

EtherCAT has five States such as Init, Pre-Operational, Safe-Operational, Operational, and Bootstrap.

RSI-ECAT-Master corresponds to the four States except Bootstrap.

The renewal of firmware performed from Bootstrap should use RSI-ECAT-Studio (option).

#### **Slave**

In EtherCAT, it means the secondary system. That is to say, it does data transmission and reception, does arbitrary operations and responses to the main control system.

It connects as follows; two or more slaves to one master. (maximum : 65535)

#### **Slave driver RSL (SDR)**

It is one of the RSI-ECAT-Master original functions.

There is library file (RSL) function so that it might be easy to treat a slave specific function, RSI-ECAT-Master is automatically loaded when there is the correspond SDR.

#### **Safe-Operational**

It is the State to which only the input of SDO communication and PDO communication is performed by one of the ESM(s). It is the State which mainly performs the check before the operation.

#### **Node**

It is the unit or the name which specify the target INtime kernel. By INtime default, it usually runs as NodeA.

#### **Byte**

It is the information unit which puts together 8 bits.

#### **Bit**

It is the information unit and is the minimum unit of data in which 1 or 0 is contained.

#### **Bootstrap**

It is the State for performing firmware downloading of a device and is one of ESM. Only the FileAccess over EtherCAT (FoE) protocol of SDO communication is available. This State is supported only by RSI-ECAT-Studio (option).

#### **Pre-Operational**

One of ESM, it is a state in which you can only do SDO communication. It is a state to initialize.

#### **Process Data**

It means that the data object including application objects designed to transfer by cyclic or non-cyclic.

#### **PDO(Process Data Object)**

It is a data structure which is defined by mapping parameter and includes the substance of one or more process data.

#### **Process data communications (PDO communications)**

It is one of the EtherCAT communications and it does cyclic communication or non-cyclic communication

#### **Master**

It is the main system which manages / constitutes the interface for the control system and which controls data transfer on a network.

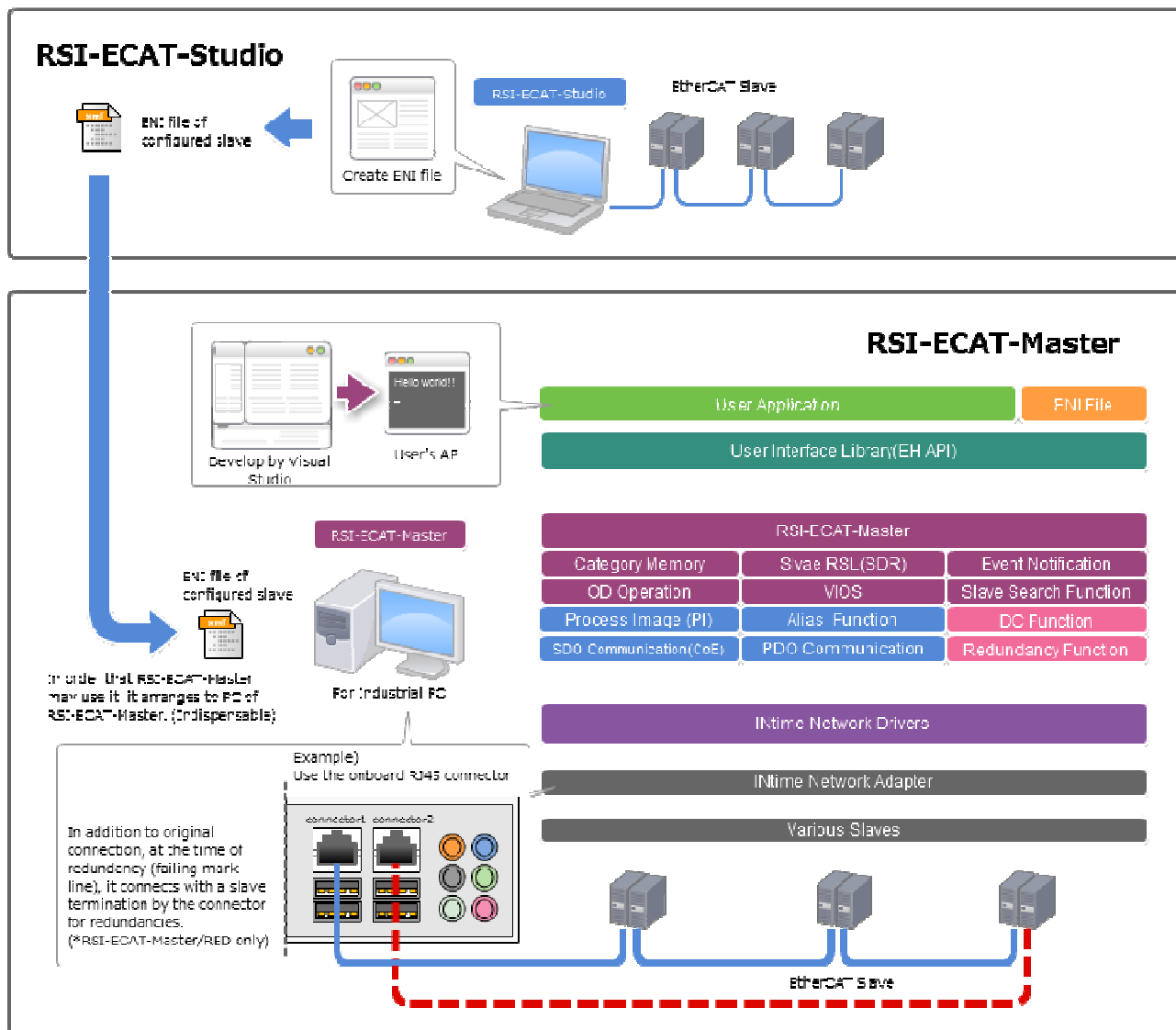
#### **Mapping Parameters**

It means the aggregate of values which define relations of application objects and process data objects.

#### **Daisy chain**

It means the method of connecting with EtherCAT slaves in an serial form.

## 6 Functional Constitution Figure



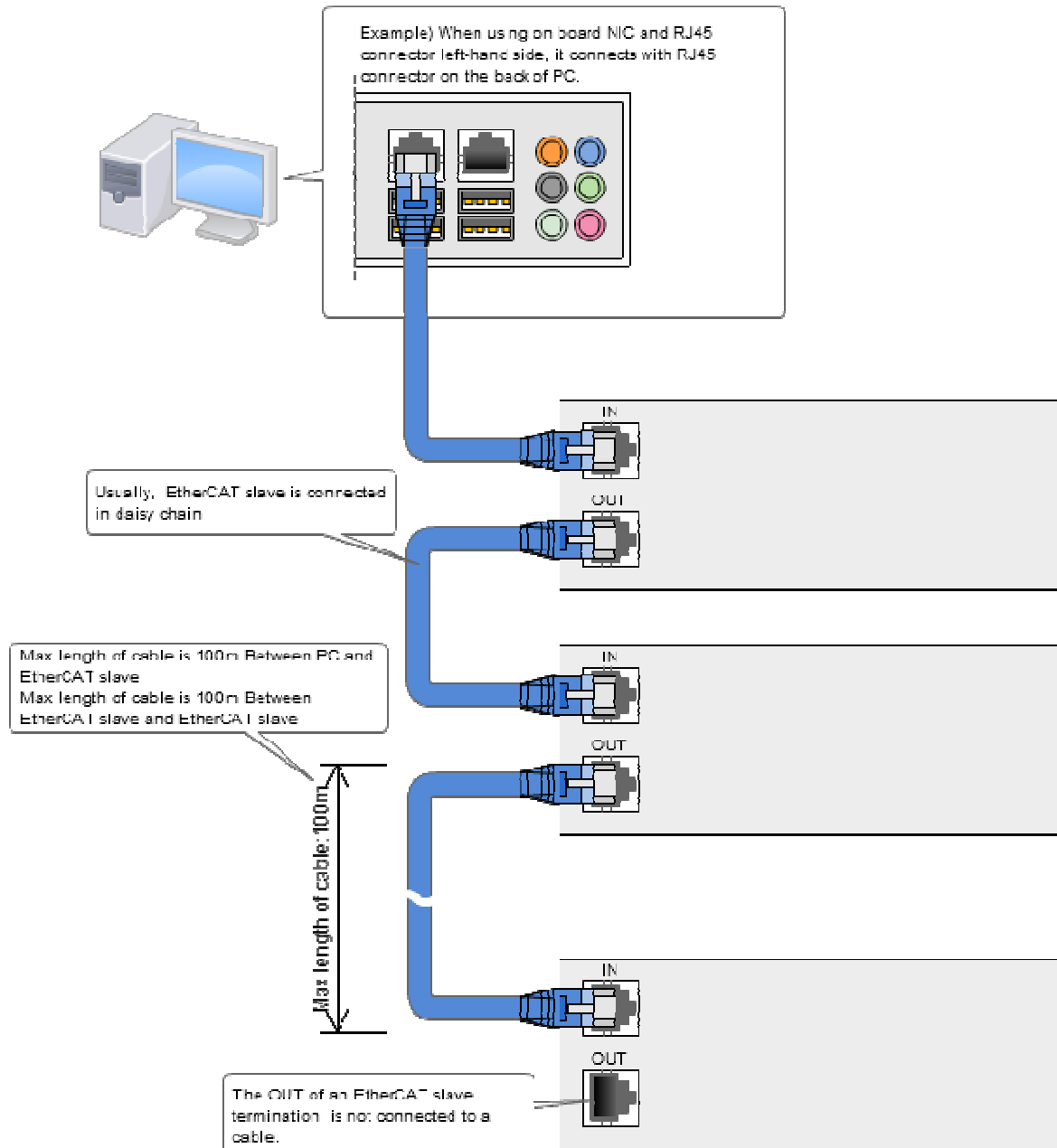
## 7 Connection of Slave Apparatus

You connect CAT5 or more Ethernet cable with the network adaptor which an EtherCAT master uses.

A network input terminal and output terminal exist in an EtherCAT slave.

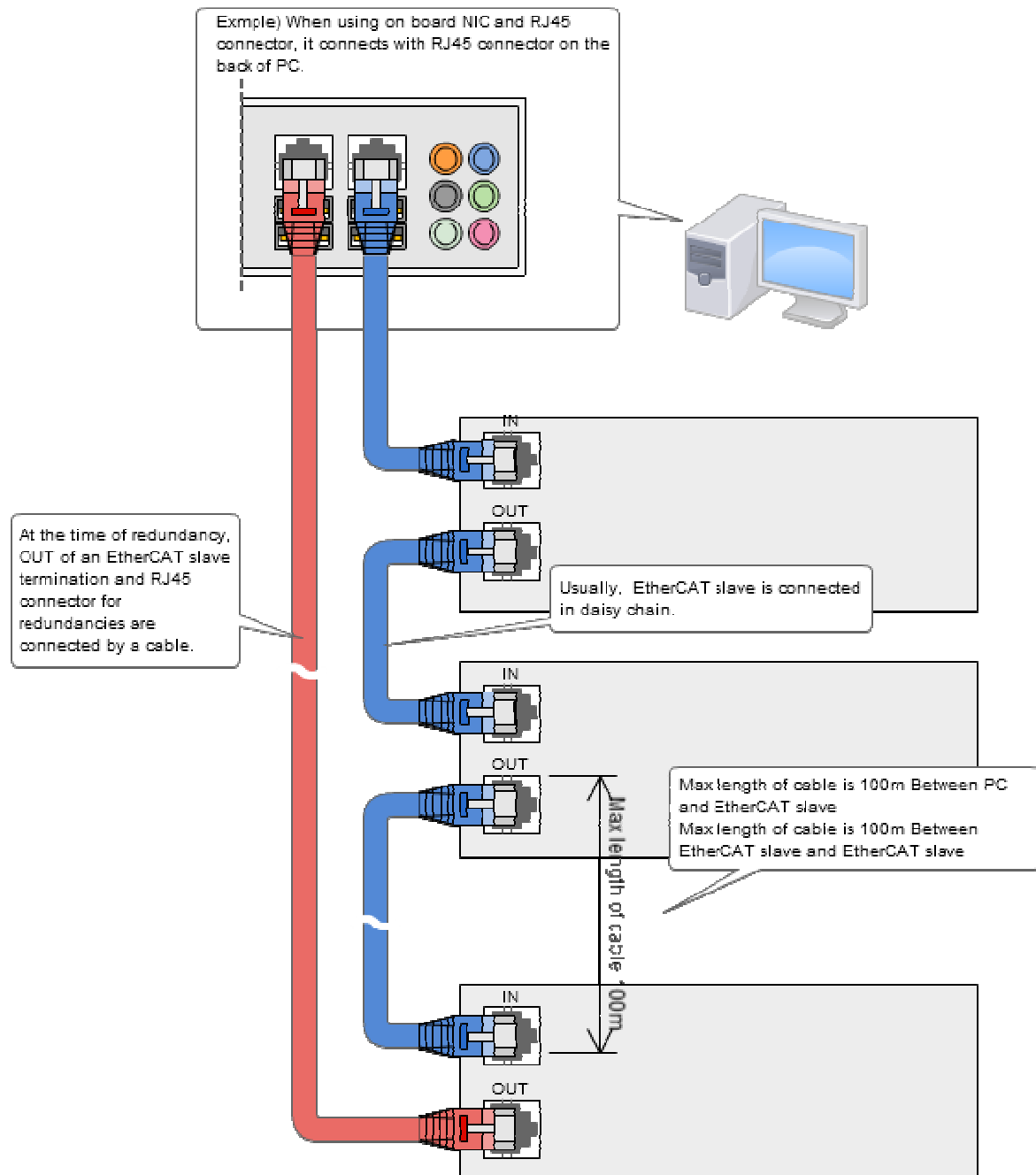
You connect the Ethernet cable from an EtherCAT master with the input terminal, and connect the input terminal from the following EtherCAT slave with the output terminal.

You don't connect anything with the trailer (=termination) of the EtherCAT slave because there is no following EtherCAT slave.



## 8 Connection in Redundancy

In addition to [ [Connection of Slave Apparatus](#) ], when you use redundancy, you connect the network adapter which EtherCAT Master uses as redundancy with the output terminal of the EtherCAT slave by an Ethernet cable which is CAT5 or more .



## 9 Requirements for Operation

This RSI-ECAT-Master product is manufactured so that it may operate on the real-time extension INtime for Windows. Moreover, RSI-ECAT-Studio products (option) are manufactured so that it may operate in the Windows environment.

You can install these 2 software products both in the same computer and in the discrete computers.

Please confirm whether your environment meets the following requests before installing.

### ■ Requirements of operation

#### 【Platform】

Microsoft Windows XP SP3 (x86) or Microsoft Windows 7 (x86/x64)

INtime 4.2

.NET Framework 2.0 (For RSI-ECAT-Studio (option))

#### 【Networked device】

Network devices that support INtime real-time network HPE2 driver

Intel 100Mbps Network adaptor (recommendation)

Intel 1Gbps Network adaptor (recommendation)

RealTek 1Gbps Network adaptor

RealTek 100Mbps Network adaptor



※ Regarding the current status of INtime HPE2 driver support,  
please refer to the following URL. (Japanese)

[http://www.mnc.co.jp/INtime/user/intimedev.php?cmd=read&page=INtime%A4%C7%BB%C8%CD%D1%B2%C4%C7%BD%A4%CANIC.4#content\\_1\\_11](http://www.mnc.co.jp/INtime/user/intimedev.php?cmd=read&page=INtime%A4%C7%BB%C8%CD%D1%B2%C4%C7%BD%A4%CANIC.4#content_1_11)

## 10 RSI-ECAT-Master Configuration File

RSI-ECAT-Master consists of the following four kinds of file.

- 1) EtherCAT Protocol Driver RSL
- 2) INtime process RTA to perform the handshake with EtherCAT protocol.
- 3) Various drivers RSL of EtherCAT slave
- 4) API-RSL

### 10.1 EtherCAT Protocol Driver RSL

EtherCAT protocol driver RSL (ecatmkpa.rsl) is used to mount an EtherCAT protocol.

This driver RSL refers to hpe2drv.rsl and an ecatmkpa.ini file in order to use NIC.

### 10.2 INtime process RTA to perform the handshake with EtherCAT protocol

The INtime process RTA (EcatHandler:RtEcHdr.rta) to perform a handshake with EtherCAT protocol uses EtherCAT protocol, established the communication and do the cyclic process.

Moreover, the user interfaces by API and a shared memory are offered, and this makes it easy to access to EtherCAT slave devices. Starting RSI-ECAT-Master means that you start this RTA (RtEcHdr.rta).

### 10.3 EtherCAT slave various drivers RSL

Various drivers RSL of EtherCAT slaves are device drivers which are for each device of connected EtherCAT slave. These files need to be arranged at the same folder where RtEcHdr.rta is.

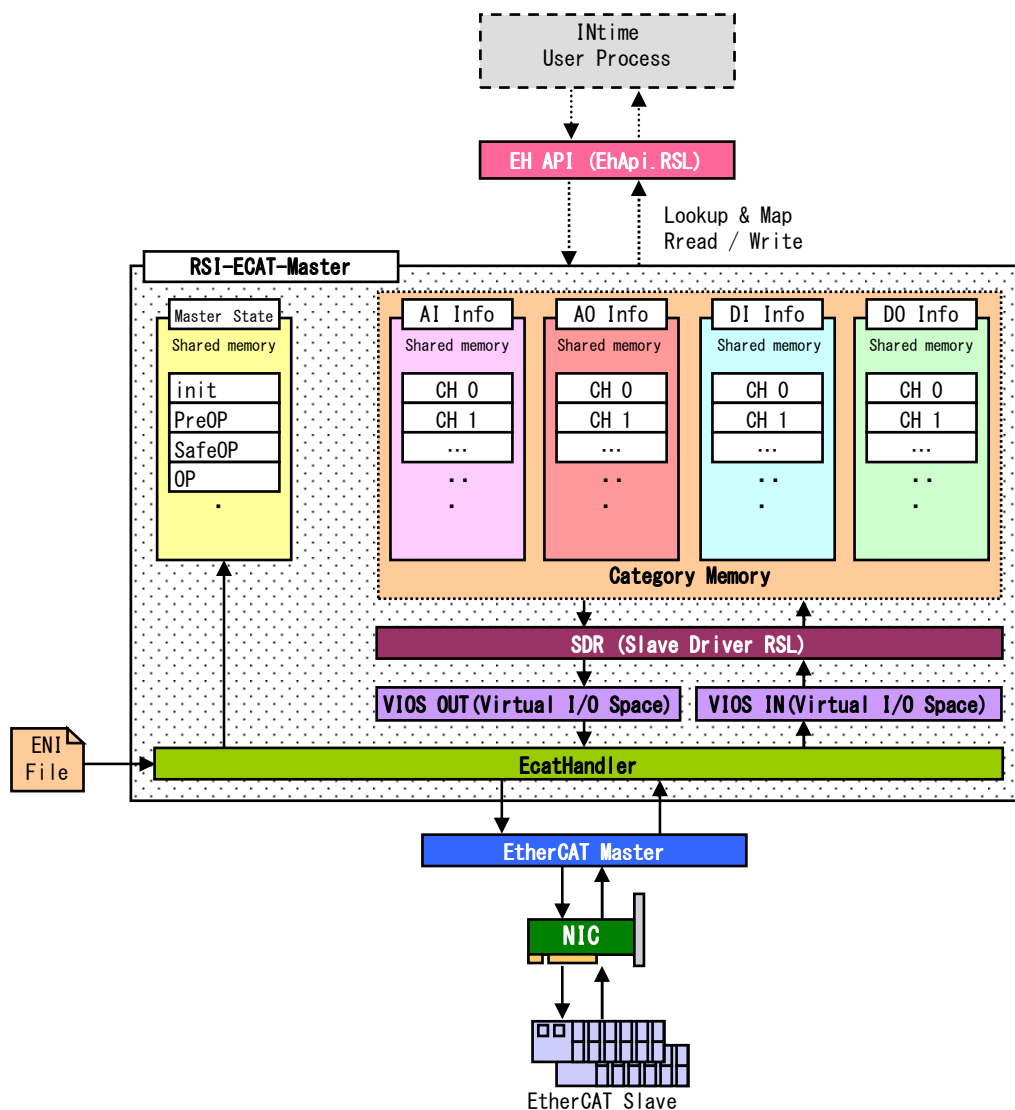
### 10.4 API-RSL

In API-RSL (EhApi.rsl), the method of accessing to EtherCAT slave devices, the method of acquiring data from EtherCAT slave devices, and the method of controlling EtherCAT State machine are offered.

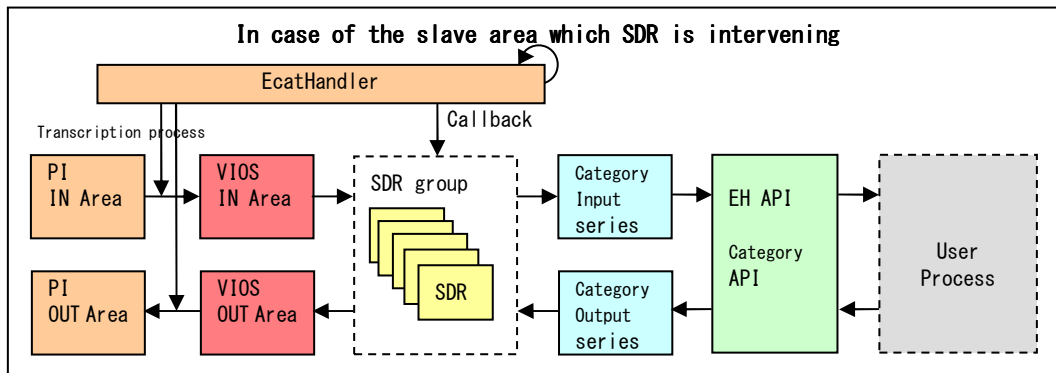
## 11 RSI-ECAT-Master configuration

RSI-ECAT-Master consists of the processes which use EtherCAT Master protocol such as EcathHandler, SDR which EcathHandler manages multiple category memories, SDR which is processed per slave (=Slave Driver RSL), virtual I/O space VIOS (=Virtual I/O Space).

All input values of PI (Process Image) per EtherCAT Master cyclic process are reflected, each SDL is called. After calling each SDL, the output values are copied from VIOS to PI of EtherCAT Master. And then EtherCAT Master sends data to each slave.

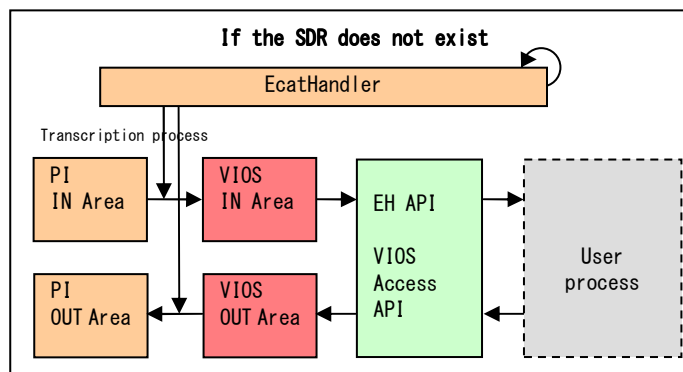


In case of the slave area which SDR is intervening, you can input and output by using category APIs from user processes since SDR read and write between category memory and VIOS.



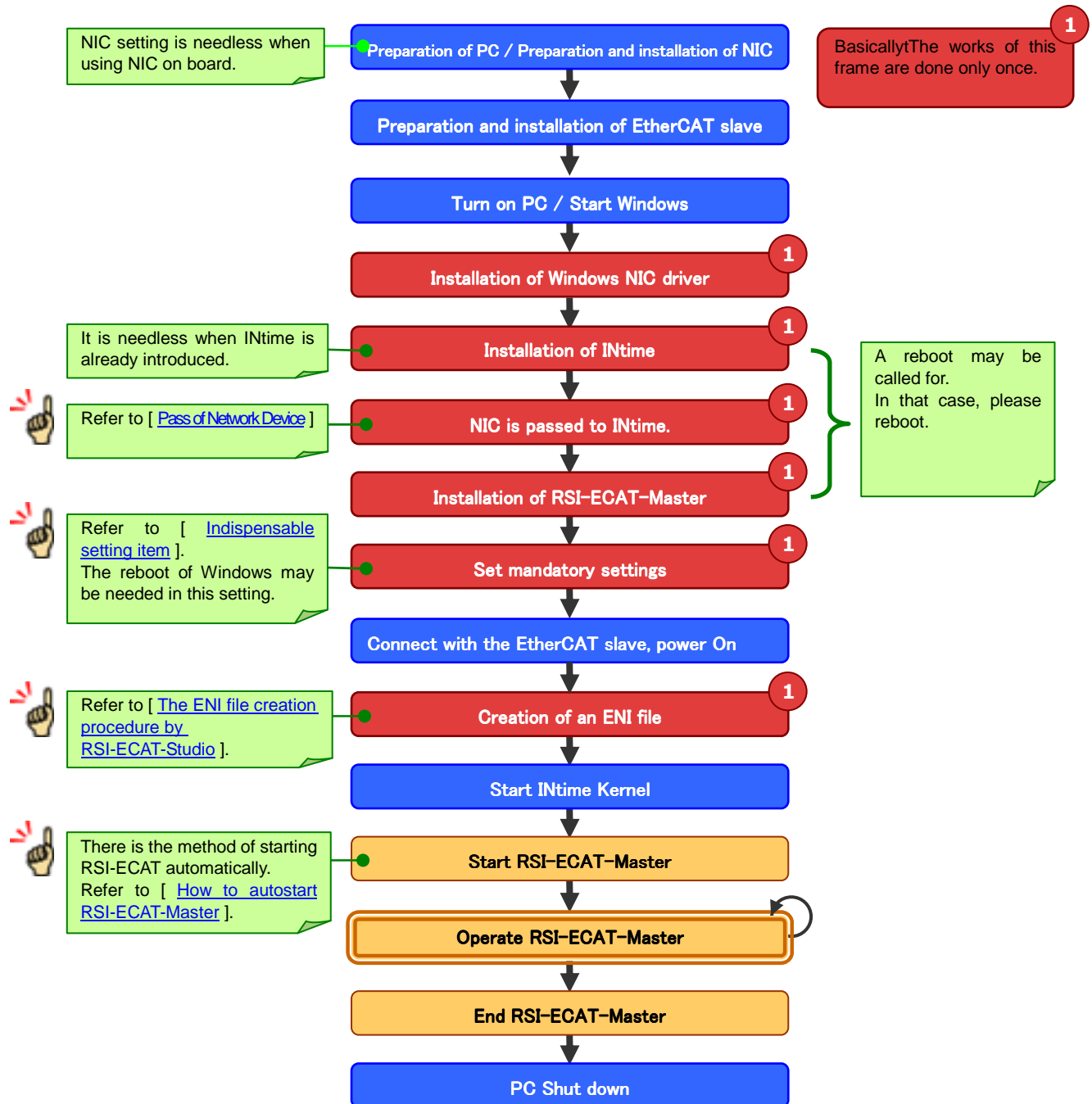
In case of the slave area which SDR isn't intervening, you have to access to VIOS directly since the value of VIOS is not acquired from category and is not moved to category.

In this case, you can input and output not by using category API but by using VIOS access API.



## 12 Start procedure flow

The next figure is a flow from PC start to an end of RSI-ECAT-Master operation.



## 13 Indispensable setting items



In order to start RSI-ECAT-Master, it is necessary to set up the following items.

### 13.1 Create ENI (EtherCAT Network Information) File

In order to use RSI-ECAT-Master, the file (ENI file) which constituted the slave is needed.

In order to make this file, RSI-ECAT-Studio (option) is used.

Refer to [ [The ENI file creation procedure by RSI-ECAT-Studio](#) ] for the procedure of creating an ENI file.

Please overwrite the created ENI file to (RSI-ECAT-Master installation path) \bin\NodeA. When you make it operate except the INtime kernel NodeA (e.g. NodeB, NodeC, etc.), please arrange the created ENI file to the folder of the node name based on [ [Specification of Start Node](#) ].

The name of an ENI file is fixed as **RtEcHdr.xml**. (half-width alphabetic character/ no case sensitivity)

### 13.2 An INtime memory is set as 64 MB or more.



RSI-ECAT-Master should be set as **64 MB or more** on INtime memory in order to use at least **64 MB** of INtime memory.

### 13.3 Specification of Start Node

※ Since the local node (NodeA) is set up by the default at the time of installation, this work is unnecessary.

When you'd like to start RSI-ECAT-Master except on INtime local node, it is necessary to set this for INtime node.

Please create the folder whose name is the same as INtime node name under the folder in which RtEcHdr.rta used as the main part of RSI-ECAT-Master exists. Then please allocate the following files to the folder.

- (1) **ecatmkpa.ini**
- (2) **RtEcHdr.xml**

### 13.4 Specification of NIC(Network Interface Controller) to be used

Please pass NIC which is used in RSI-ECAT-Master to INtime by the procedure of [ [Pass of Networked Device](#) ]. Then, it is necessary to specify the INtime NIC driver used for ecatmkpa.ini.

It is specified as the **HPE2Ifs** key in the **KPAMasterINtime** section of ecatmkpa.ini. ([ecatmkpa.ini setting item](#) )]

You define it as INtime NIC driver name + instance number (example: ie1g0).

There are four kinds of INtime NIC drivers which can be specified. Please choose according to NIC.

- (1) **ie1g** - Intel 1Gbps Series Network device
- (2) **ie100m** - Intel 10/100Mbps Series Network device

- (3) **rtl1g** – RealTek 1Gbps Series Network device
- (4) **rtl100m** – RealTek 10/100Mbps Series Network device

An instance number is regarded as an identification number for NIC when two or more same NICs exist on the PC.

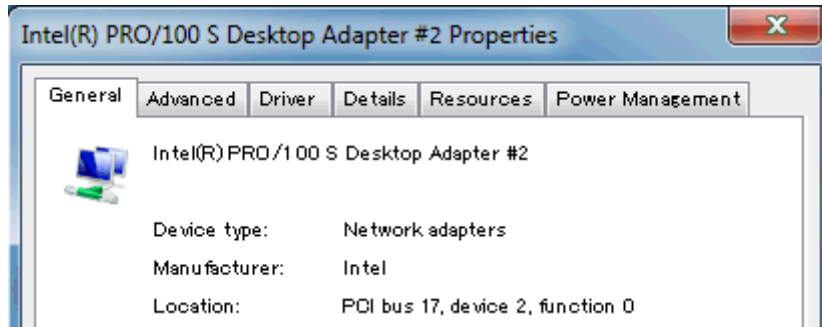
The detected order in PCI configuration space among NICs passed to INtime is assigned from 0.



※ Orders that PCI devices is detected from PCI configuration space are not based on their slot positions, memory arrangements, etc, but based on PCI bus numbers, device numbers, function numbers. Please check these numbers by Windows device manager etc.

【Example】

① PCI – Intel PRO/100 S Bus number[17] Device number[2] Function number[0]



② PCI – Intel PRO/100 S Bus number[5] Device number[2] Function number[0]



In this case,

the device of ② “PCI Bus 5, device 2” is applied to the instance 0.

The device of ① “PCI Bus 17, device 2” is applied to the instance 1.



※ Please do not set up NICs used by RSI-ECAT-Master as in INtime Network 7 and INtime Legacy network. As for the NIC drivers used by RSI-ECAT-Master, RSI-ECAT-Master boots.

### 13.5 Mandatory alias



RSI-ECAT-Master certainly uses an alias since a slave is specified by API etc..

From specification of a slave becoming impossible If aliases overlap, since it is impossible to specify a slave, so an error message is displayed and operation is interrupted. Please set up it not to overlap aliases. When you'd like to set up an alias without using Station Alias, please use RSI-ECAT-Studio (option).

Refer to [ [Alias setup steps by RSI-ECAT-Studio](#) ] for a setup of alias by RSI-ECAT-Studio (option).

### 13.6 Station Alias = Alias



RSI-ECAT-Master uses this value as an alias when the value of the ESC register (0x0012 Station Alias) of each slave is other than "0". When the value of Station Alias differs from the alias number currently recorded on EEPROM, it is recorded as the alias number of Station Alias on EEPROM. (\* when the value of Station Alias is "0", it isn't written in.)

It may be linked with rotary switches or DIP switches etc. or which are mounted on the ESC register (0x0012 Station Alias) according to EtherCAT slaves.

Please refer to EtherCAT slave vendors on whether the ESC register (0x0012 Station Alias) is used or not.

The value of an ESC register (0x0012 Station Alias) can be checked from RSI-ECAT-Studio (option). Open Memory of an applicable slave to check "0x0012".

The screenshot shows the RSI-ECAT-Studio interface with the 'Memory' tab selected. The 'EEPROM' sub-tab is active. The 'Start offset (in bytes)' is set to 0x0 and the 'Length (in bytes)' is 512. The 'Format' is set to DEC. The table below shows the memory contents:

Offset	Name	Value (Dec)	Value (Hex)
000E		0	00
000F		0	00
0010	Configured Station Address	1001	03E9
0012	Configured Station Alias	255	00FF
[15:0] Alias Address used for node addressing (FP...		255	00FF
0014		0	00
0015		0	00

In this example, 255 is assigned.

## 14 Pass of Network Device

One network device is used in RSI-ECAT-Master. (Two network devices are used in redundancy for entrance and exit. Although a network device is usually Windows management, it is necessary to pass this to INtime in order to make it use from RSI-ECAT-Master.

### (1) Addition of network device

Install the network device used by RSI-ECAT-Master to the PC in which RSI-ECAT-Master is installed. This work is unnecessary when RSI-ECAT-Master is using the network device mounted in PC. Please refer to [ [Requirements for Operation](#) ] for the network device corresponding to RSI-ECAT-Master.

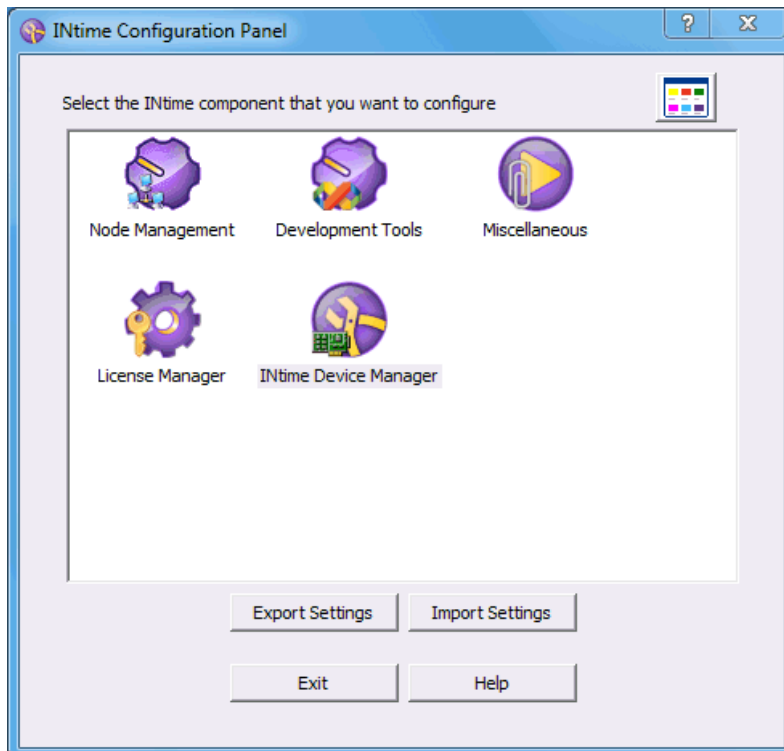
### (2) Installation of a network device driver

When a network device is newly added or installed, installation of a Windows device driver is called for after Windows starting. Please install a Windows device driver according to a device driver wizard.

### (3) Pass to INtime

INtime needs to be installed.

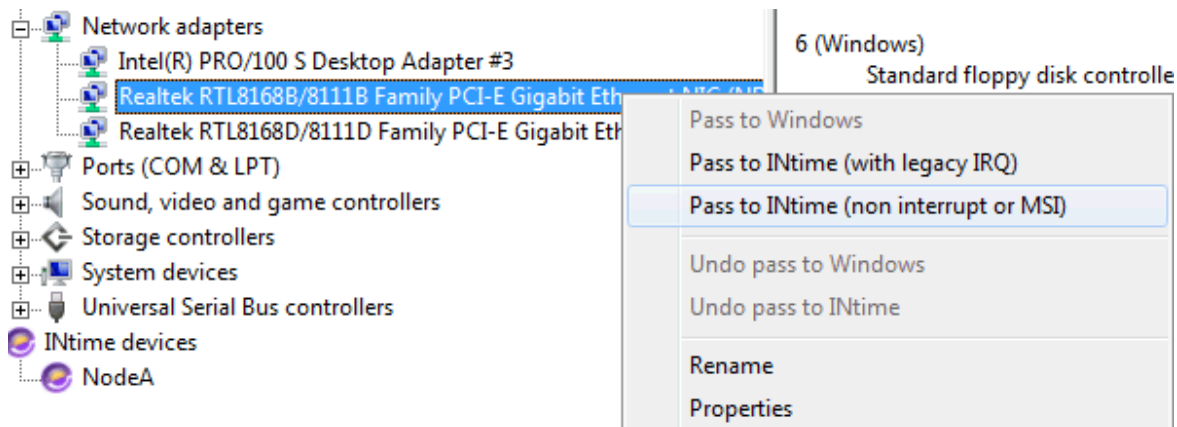
Click 「All Programs」 -> [INtime] -> [INtime Configuration] of the Start Menu, [INtime Configuration Panel] will be displayed. Double-click "INtime Device Manager" icon from the list of INtime Configuration Panel.



(4) Select a network device that you want to pass to INtime in order to use in RSI-ECAT-Master.  
Please perform either (A) or (B) according the type of the network device.

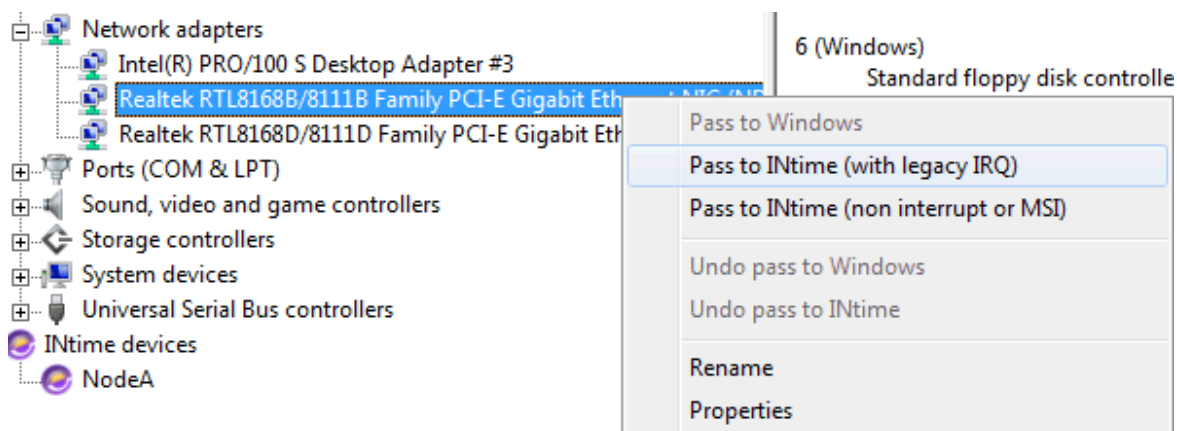
(A) In case of the network device of PCI Express type with MSI support:

Please choose [Pass to INtime (non Interrupt or MSI) ] from the pop up menu of a right-click, and click.



(B) In case of the network device of PCI type :

Please choose [Pass to INtime (with legacy IRQ) ] from the pop up menu of a right-click, and click.



※ If "INtime CONFLICT!!!" is displayed in the right pane after performing (B)

You cannot use this since IRQ is competing with Windows device(s).

Please interrupt this work, and repeat all the Windows devices that are competing, or change the PCI slot position of the network device, then try again.

RSI-ECAT-Master cannot be used in this state.

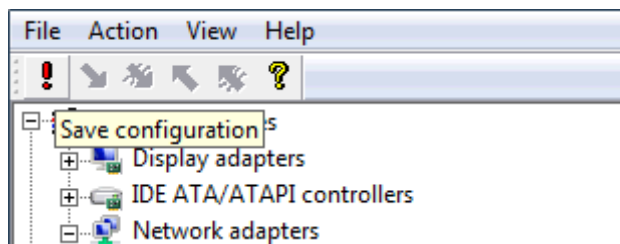
18 (Windows)

Intel(R) PRO/100 S Desktop Adapter #3  
Intel(R) 5 Series/3400 Series Chipset Family PCI Express Root Port 3 - 3B4  
Intel(R) 5 Series/3400 Series Chipset Family PCI Express Root Port 7 - 3B4  
Intel(R) 5 Series/3400 Series Chipset Family USB Enhanced Host Control  
Intel(R) 5 Series/3400 Series Chipset Family USB Universal Host Controll  
Intel(R) 5 Series/3400 Series Chipset Family USB Universal Host Controll

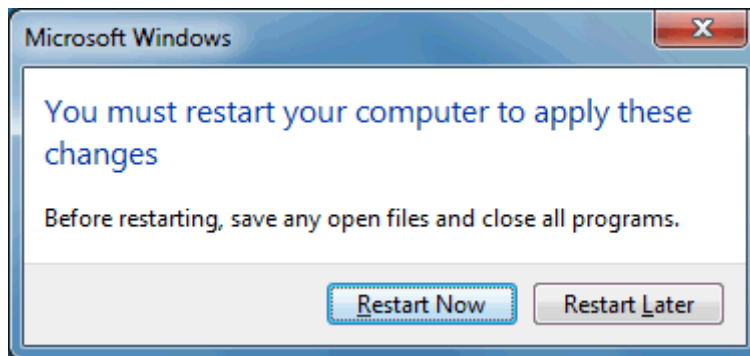
-> (INtime) CONFLICT!!!

Realtek RTL8168B/8111B Family PCI-E Gigabit Ethernet NIC (NDIS 6.20)

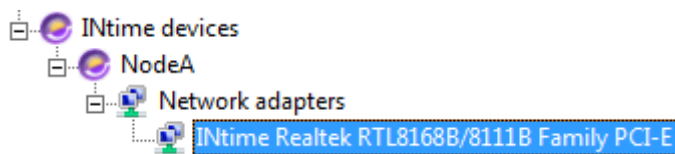
(5) Click [ ! ] icon of tool bar and make it apply to the path to INtime.



(6) The reboot of Windows is called for depending on the environment. In this case, please perform the reboot of Windows.



(7) The network device will have been moved to the tree of [INtime devices] on [INtime Device Manager] after finishing to pass to INtime.



(8) By doing the above, passing network device finishes.

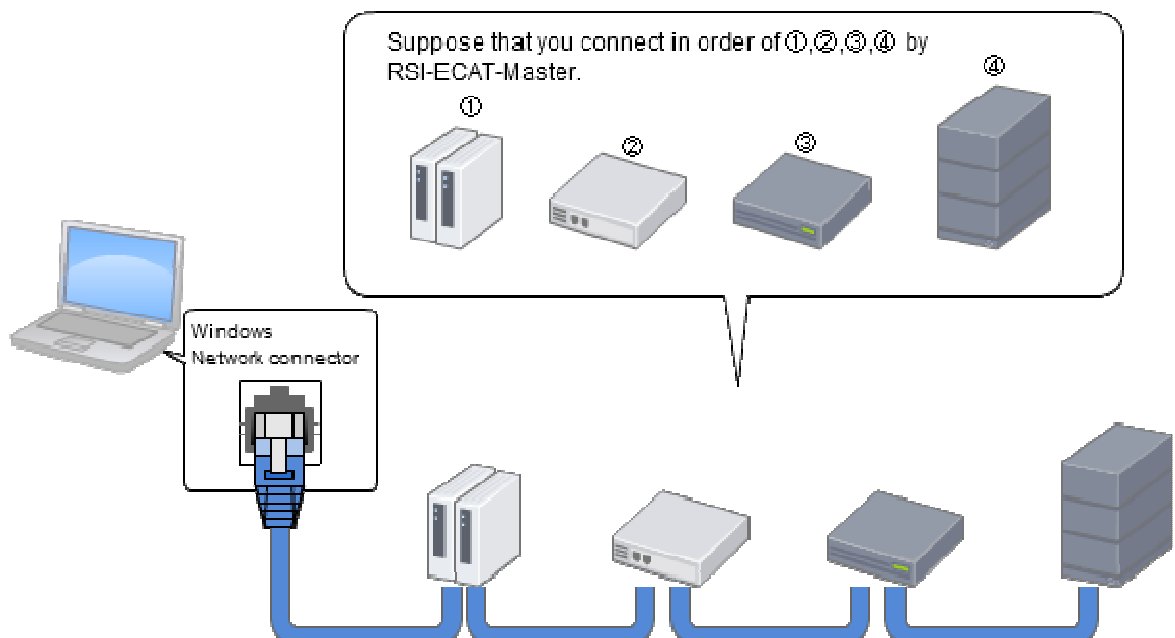
## 15 The ENI file creation procedure by RSI-ECAT-Studio

### 15.1 The things to prepare

- A) PC where RSI-ECAT-Studio (option) has been installed and where Windows network (cable) can be used.
- B) EtherCAT slave (groups) which you'd like to connect by RSI-ECAT-Master
- C) Over CAT5 Ethernet cable to connect between an EtherCAT slave and PC (less than 100m)
- D) Over CAT5 Ethernet cable to connect between an EtherCAT slave and other EtherCAT slave (less than 100m)
- E) Space to store an ENI file (recording media such as a USB memory)
  - \* Use it for moving an ENI file to PC of RSI-ECAT-Master.
- F) ESI file for B) ※ The ESI file is offered from the EtherCAT slave vendor.

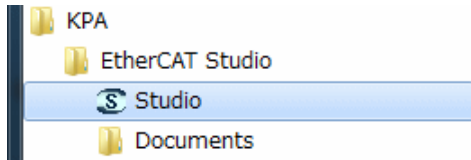
### 15.2 Pre-Preparation

- ① Using A) and C), connect IN connector of EtherCAT slave which is the head of B) with the connector for Windows network.  
Using B) and D), do daisy chain connection of the EtherCAT slave by the line used on RSI-ECAT-Master.  
Please connect directly between A) and B) without HUB this time.
- ② Then, the power supply of each slave is turned ON.
- ③ Connect the recording media of E) to PC.

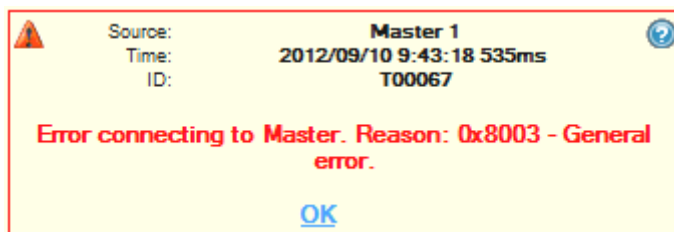


### 15.3 Application of ESI File

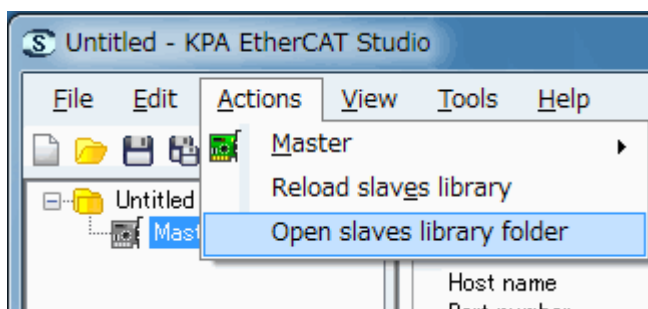
- ④ Click 「All Programs」 -> [KPA] -> [EtherCAT Studio] -> [Studio] of the Start Menu, RSI-ECAT-Studio will be started.



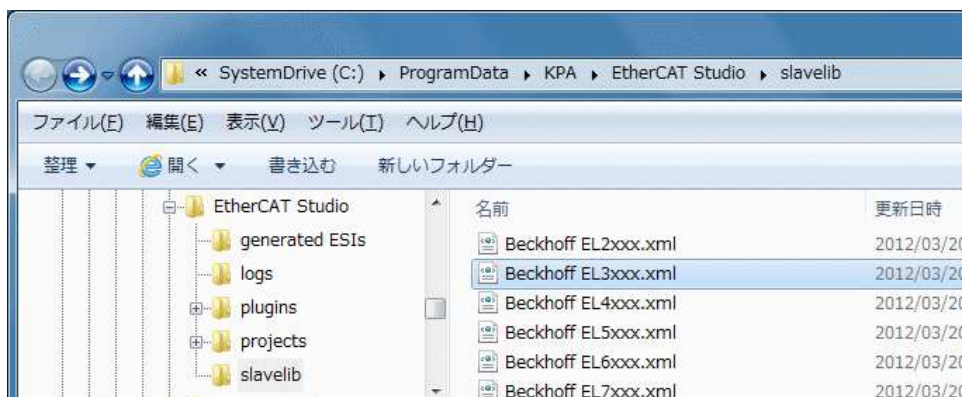
- ⑤ If the following errors are displayed on a RSI-ECAT-Studio start-up, EtherCAT slave(s) are not been connected. At the present stage, please push OK and close an error because you will do the check of the connection later.



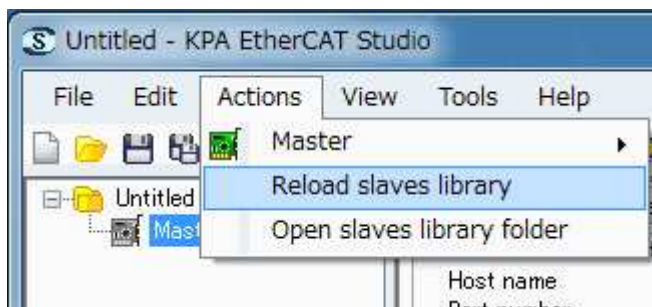
- ⑥ A KPA EtherCAT Studio window is opened.  
Click [Actions] -> [Open slaves library folder] of the menu bar.



- ⑦ Explorer is opened and the path in which the ESI file is stored is displayed.  
Please arrange here the ESI file(s) prepared by F).

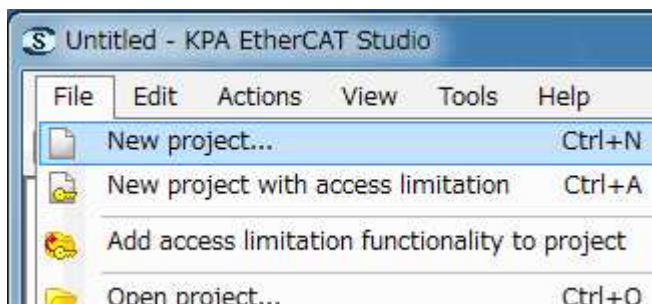


- ⑧ Close the Explorer after ⑦. Then return to a KPA EtherCAT Studio screen, click [Actions] → [Reload slaves library] of the menu bar to make ESI file recognize.



#### 15.4 Creation of ENI File

- ⑨ Click [File] → [New project...] of the menu bar to create a new project.

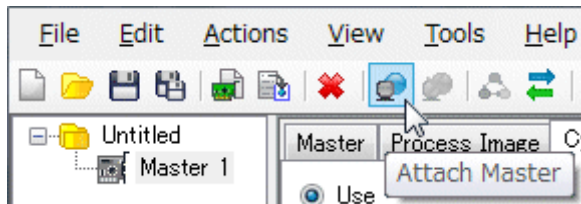


- 
- Untitled\* - KPA EtherCAT Studio
- File Edit Actions View Tools Help
- Untitled
- Master 1
- Master Process Image Cyclic Distributed Clocks
1. Connection settings for master server
- |             |           |
|-------------|-----------|
| Host name   | localhost |
| Port number | 5000      |
2. Master parameters
- |                   |   |
|-------------------|---|
| Master name       | Master 1                                      |
| Network card      | 2.Realtek RTL8168D/8111D Family PCI-E GBE NIC |
| Enable redundancy | False   |

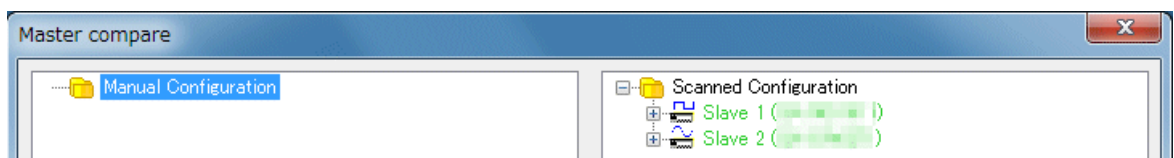
- 
- Untitled\* - KPA EtherCAT Studio
- File Edit Actions View Tools Help
- Untitled Master 1
- Master Process Image Cyclic Distributed Clocks
- ☐ Create PI by order
 ☒ Create PI by tree view
- Alignment mode: No alignment
- ☐ Show physical property values
- Output variables (size: 0 bytes, length: 0 bytes)
- | Name | Type | Bit size | PI offset |
|------|------|----------|-----------|
|      |      |          |           |

- 
- Untitled\* - KPA EtherCAT Studio
- File Edit Actions View Tools Help
- Master Process Image Cyclic Distributed Clocks
- ☒ Use LRW
- ☐ Use LRD/LWR
- ☐ Use FPRD/FPWR

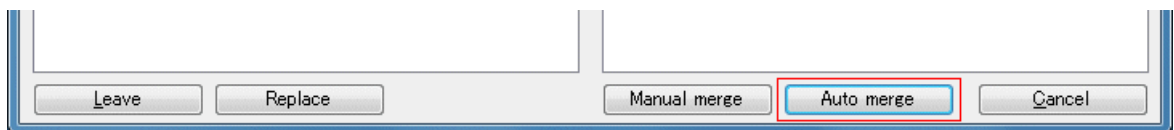
- ⑬ Click [Attach Master] of the tool bar so that EtherCAT communication starts to detect slave(s).



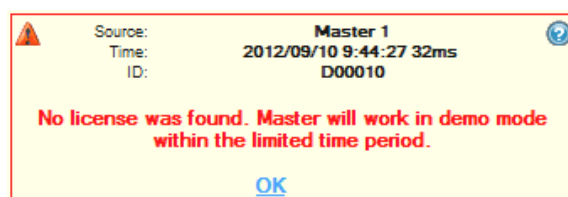
- ⑭ The [Master compare] screen is displayed in several seconds after the EtherCAT communication starts.



Please push [Auto Merge] button and reflect the detected slave information.



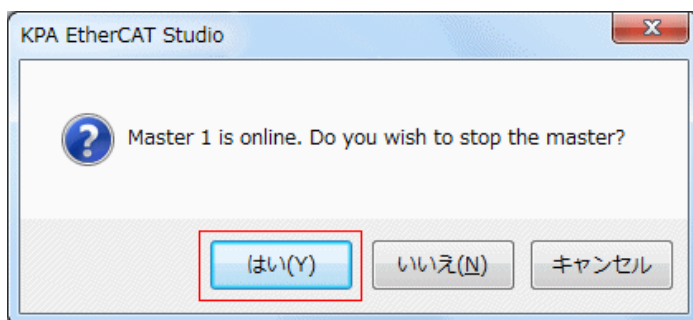
- ⑮ After [Attach Master], the following warning message is displayed.  
This procedure describes to use a demo license of EtherCAT Master for Windows in order to create ENI file. Please push [OK] and close this warning message.



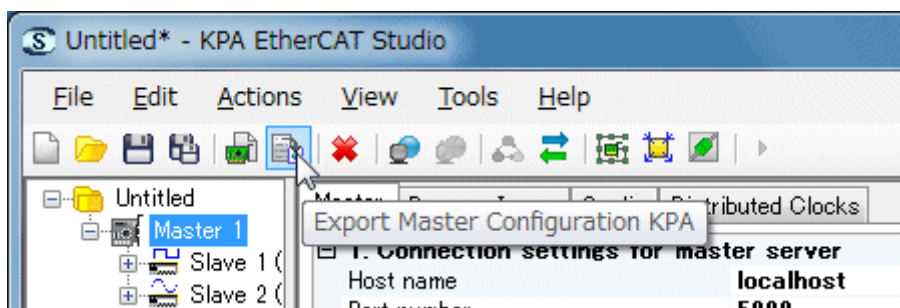
- ⑩ Click [Detach Master] of tool bar to let EtherCAT communication stop.



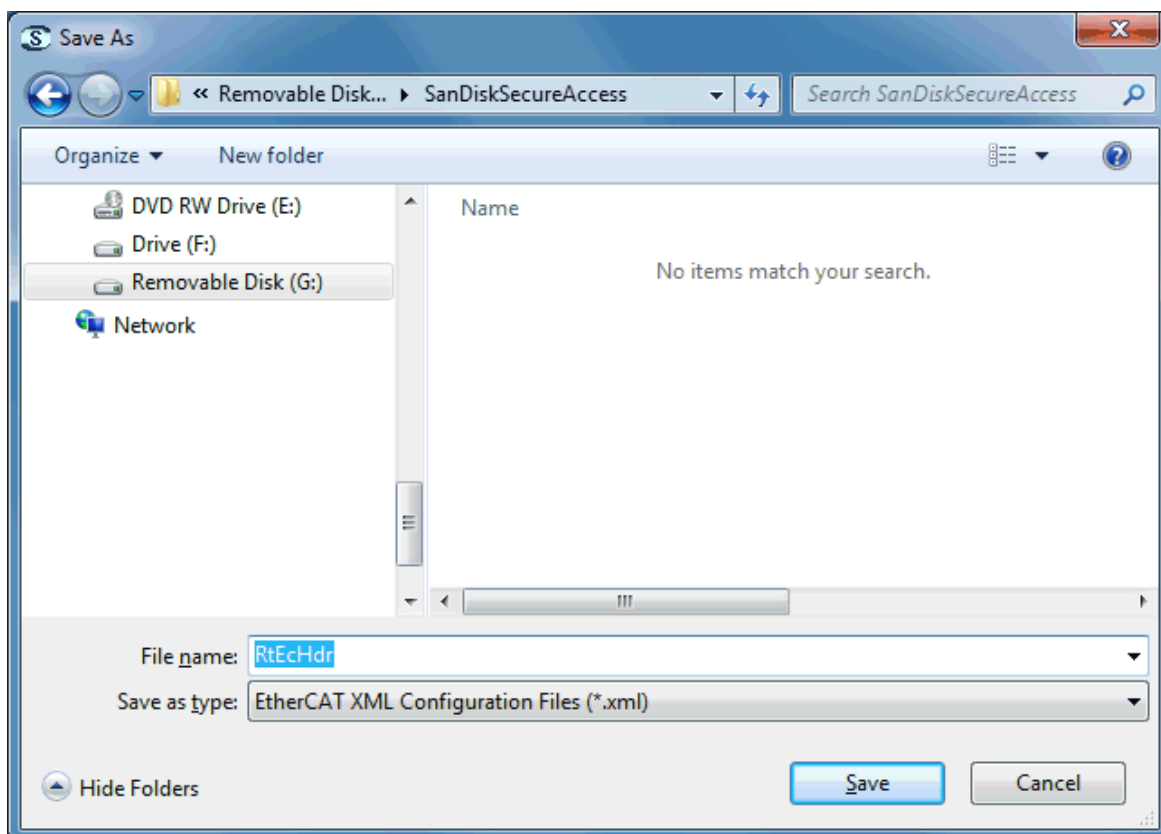
It is checked whether EtherCAT communication should be stopped or not. Please click [Yes. (Y)] and continue.



- ⑪ Click [Export Master Configuration KPA] of tool bar to let the ENI file output.



- ⑮ Set the recording-media drive of E) from the left tree view, and name [RtEcHdr] as the file name.  
 ※ This time please check that [File type (T):] is [EtherCAT XML Configuration Files (\*.xml)] .  
 Click [Save (S)] button.



#### 15.5 The ENI file created to RSI-ECAT-Master is arranged or updated.

- ⑰ The file [RtEcHdr.xml] saved at the recording media of E) is moved to PC in which RSI-ECAT-Master is installed. The default installation path of RSI-ECAT-Master is as follows.

Since I arrange the default file at the time of RSI-ECAT-Master installation, it is overwritten.

Environment	Default installation path
Windows XP/Windows 7 32bit	C:\Program Files\Micronet\RSI-ECAT\bin\NodeA\RtEcHdr.xml
Windows 7 64bit	C:\Program Files (x86)\Micronet\RSI-ECAT\bin\NodeA\RtEcHdr.xml

- ⑱ By doing the above, the ENI file creation procedure by RSI-ECAT-Studio will be completed.

## 16 Alias setup steps by RSI-ECAT-Studio

### 16.1 The things to prepare

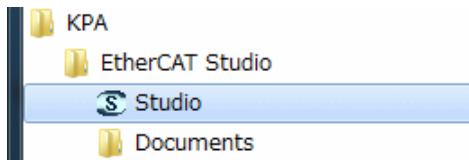
- A) PC where RSI-ECAT-Studio (option) has been installed and where Windows network (cable) can be used.
- B) EtherCAT slave group to set an alias
- C) Over CAT5 Ethernet cable to connect between an EtherCAT slave and PC (less than 100m)
- D) Over CAT5 Ethernet cable to connect between an EtherCAT slave and other EtherCAT slave (less than 100m)

### 16.2 Pre-Preparation

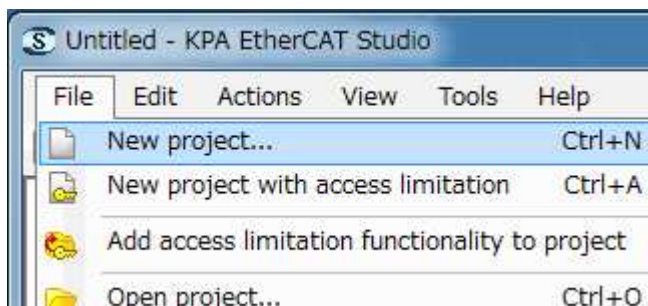
- ① Using A) and C), connect IN connector of EtherCAT slave which is the head of B) with the connector for Windows network.  
 Using B) and D), do daisy chain connection of the EtherCAT slave by the line used on RSI-ECAT-Master.  
 Please connect directly between A) and B) without HUB this time.
- ② Then, the power supply of each slave is turned ON.

### 16.3 Detection of the EtherCAT slave connected

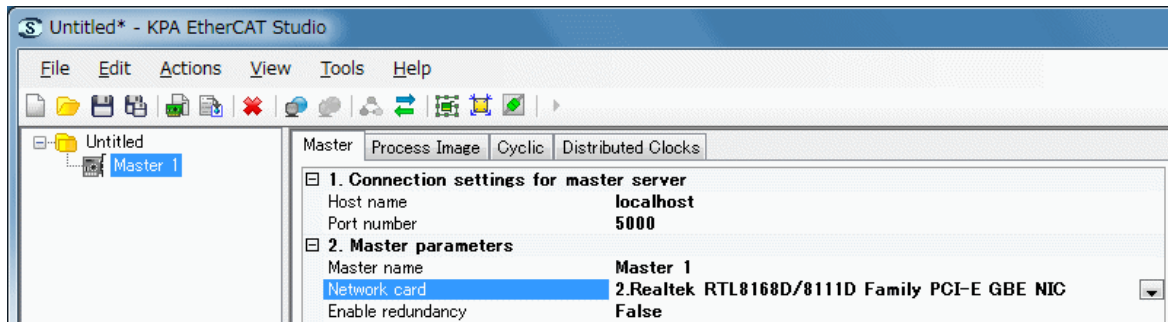
- ③ Click 「All Programs」 -> [KPA] -> [EtherCAT Studio] -> [Studio] of the Start Menu, RSI-ECAT-Studio will be started.



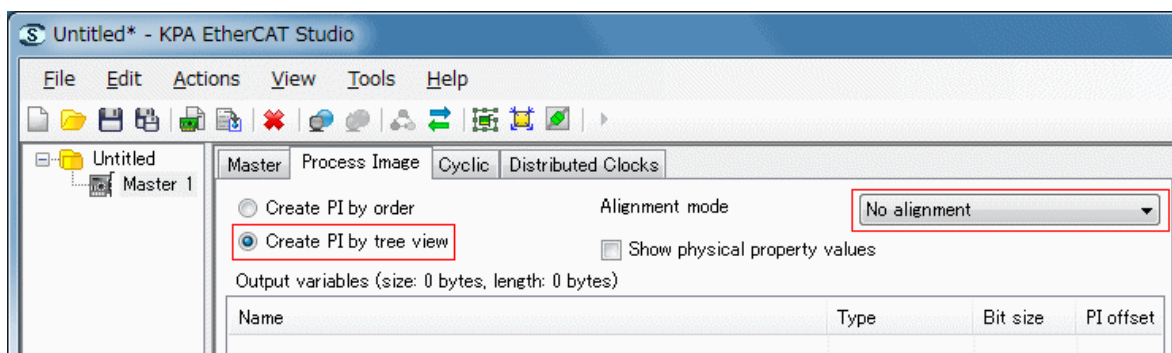
- ④ Click [File] -> [New project...] of menu bar, create a new project.



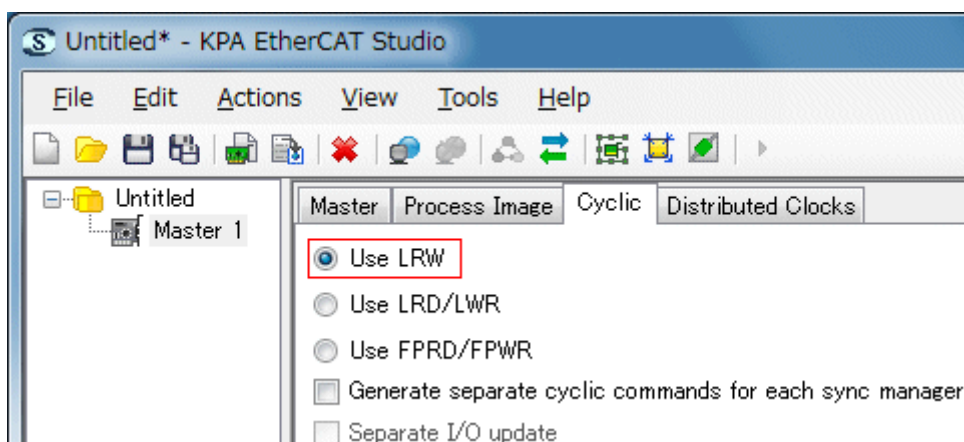
- ⑤ The [Master] tab which is on the right window after selecting [Master 1] from the left window. Please choose the [Network card] item and choose the network adaptor to which B) is connected from [Network card].



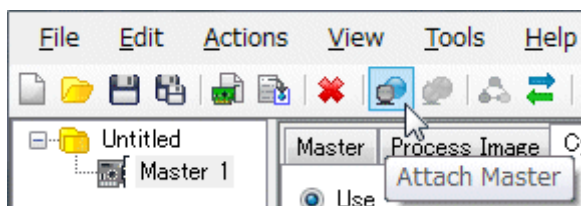
- ⑥ Select [Process Image] tab. Set [Alignment mode] as [No alignment] after selecting [Create PI by tree view]



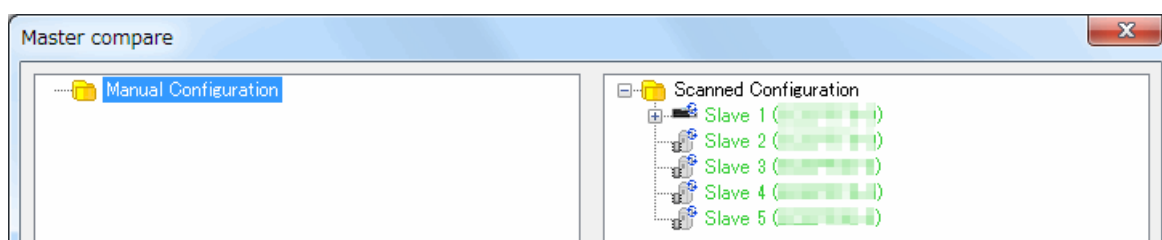
- ⑦ Select [Cyclic] tab. Select [Use LRW].



- ⑧ Click [Attach Master] of the tool bar so that EtherCAT communication starts to detect slave(s).



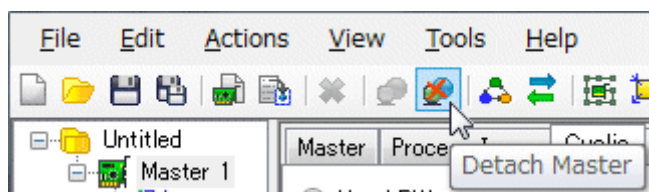
- ⑨ The [Master compare] screen is displayed in several seconds after the EtherCAT communication starts.



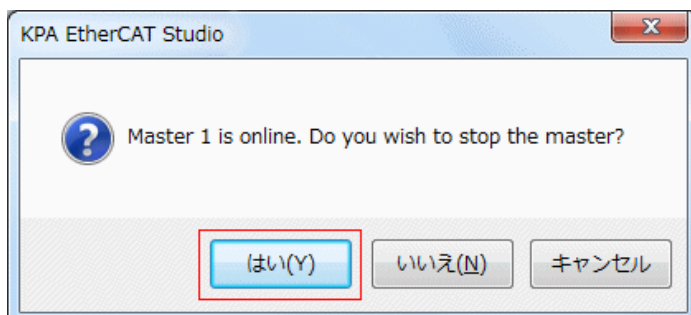
Please push [Auto Merge] button and reflect the detected slave information.



- ⑩ Click [Detach Master] of tool bar to let EtherCAT communication stop.

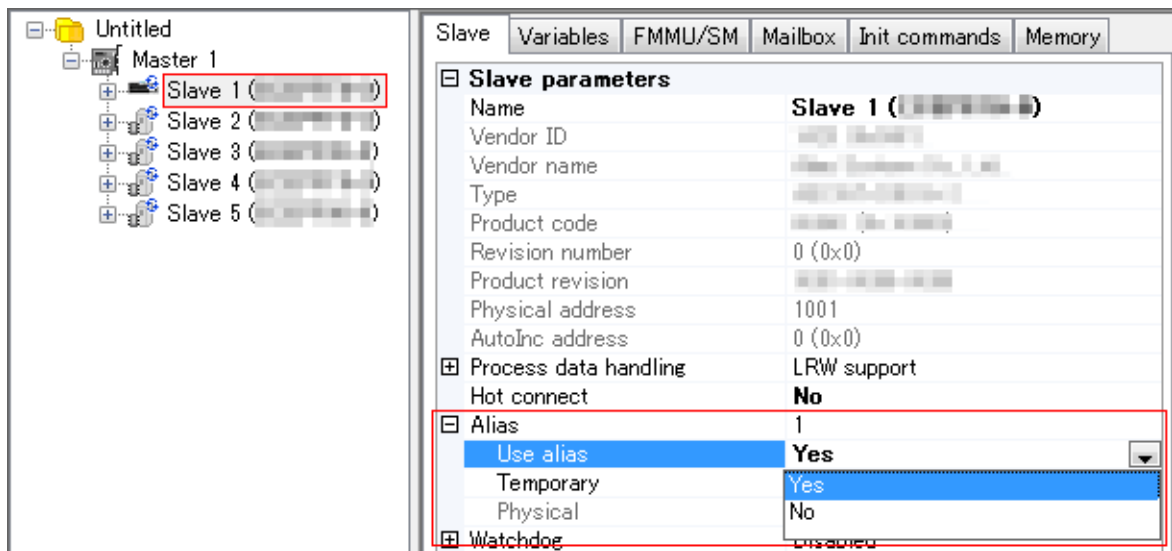


It is checked whether EtherCAT communication should be stopped or not. Please click [Yes. (Y)] and continue.



#### 16.4 Setup of temporary alias

- ⑪ Please choose a slave to set an alias after the EtherCAT communication stops, and display the setting item of [Alias] from the [Slave] tab by clicking [+] or [-] of [Alias] item. Then, please change the item of [Use alias] to [Yes].

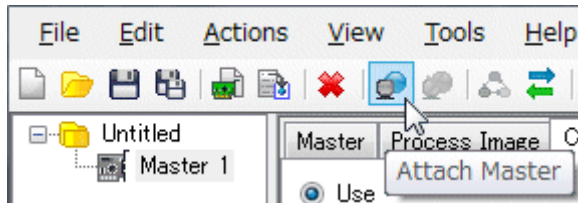


- ⑫ Input the alias number which you'd like to set to [Temporary] item as after setting [Use alias] item as [Yes]. The range which can be input is 1 to 65535. Be careful that these values are not overlapped with ones of other slaves. The value of [Physical] displayed under the [Temporary] item is the alias assigned to current slave(s).

Alias	1
Use alias	Yes
Temporary	30
Physical	1
Watchdog	Disabled

(※ 30 is set in this example.)

- ⑬ The work of ⑫ and ⑬ is similarly applied to other slaves which you'd like to set.
- ⑭ Click [Attach Master] of the tool bar let EtherCAT communication start in order to detect slave(s).



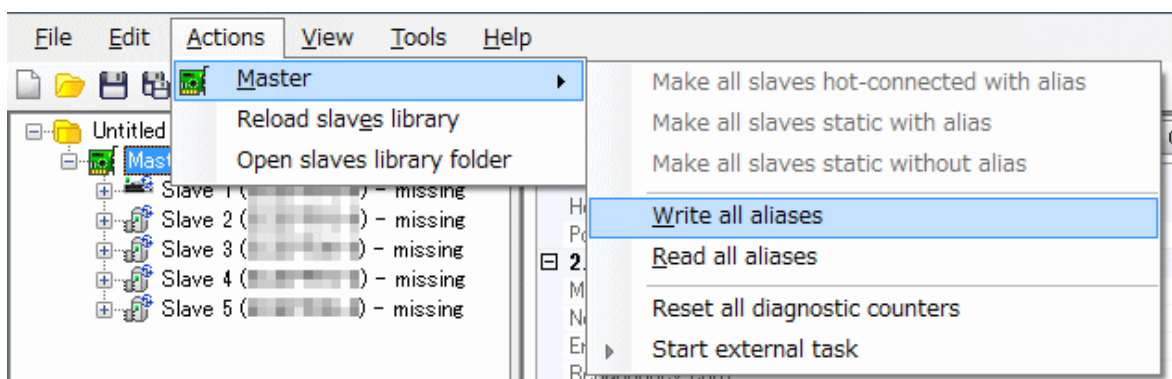
- ⑮ Some error messages may be displayed when EtherCAT communication starts. It means that the information set in RSI-ECAT-Studio doesn't correspond to the on-line information. Please skip all these errors by clicking [OK].

Example of error generating :

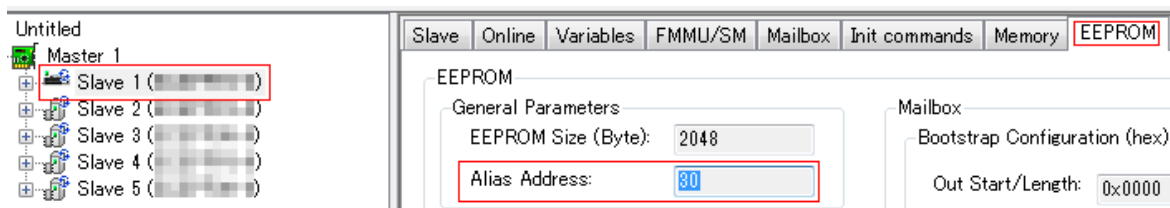


## 16.5 Applied as Regular alias

- ⑯ Click [Actions] from menu bar and click [Write all aliases] in [Master] item after EtherCAT communication starts.

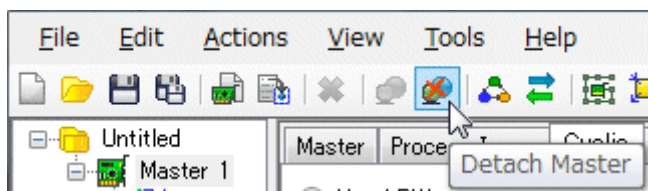


- ⑰ Choose slave and choose the [EEPROM] tab in order to check whether the alias has been applied to the slave or not. If the value set as the [Alias Address:] box is stored, it is applied normally.

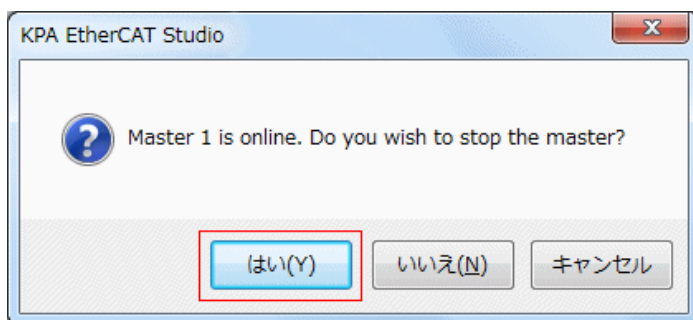


(※ 30 is set in this example.)

- ⑱ Click [Detach Master] of tool bar to let EtherCAT communication stop.



It is checked whether EtherCAT communication should be stopped or not. Please click [Yes. (Y)] and continue.



- ⑲ By doing the above, the alias setup steps by RSI-ECAT-Studio will be completed.

## 16.6 Cautions of alias setup



Even if alias is set up, when the value of [Station Alias] is other than “0”, RSI-ECAT-Master will update the value of [Station Alias] as the alias.

In order to use the alias set up with [ [Alias setup steps by RSI-ECAT-Studio](#) ] by RSI-ECAT-Master, please set the value of [Station Alias] as “0”.

For details, please refer to the [ [Station Alias = Alias](#) ] item.

Offset	Name	Value (Dec)	Value (Hex)	Description
000F		0	00	
0010	Configured Station Address	1001	03E9	
0012	Configured Station Alias	0	0000	
0014		0	00	

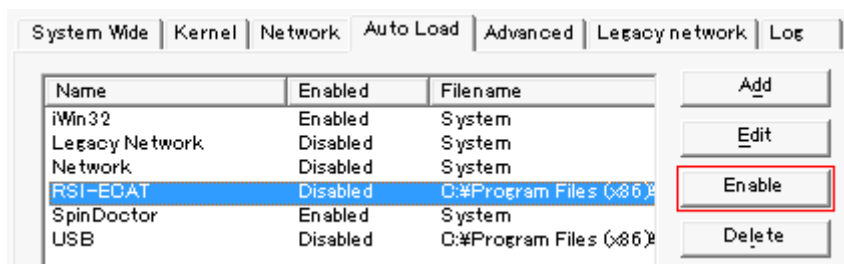
## 17 How to start the RSI-ECAT-Master

There is a file called RtEcHdr.rta in RSI-ECAT-Master as a core. Please start RtEcHdr.rta when you start RSI-ECAT-Master manually. RtEcHdr.rta is stored in the \ (RSI-ECAT-Master installation path) Micronet\RSI-ECAT\bin\ folder. Moreover, the definition file is stored in the NodeA folder.

## 18 How to autostart RSI-ECAT-Master

When RSI-ECAT-Master is installed, RSI-ECAT-Master will be registered to Autoload of INtime, but the automatic start is usually an invalid state. Please change this invalid state to valid and save it if you'd like to start RSI-ECAT-Master together with INtime kernel start.

Regarding the Autoload item of INtime, open Node Managements of INtime Configuration Panel firstly. After selecting NodeA of an INtime Node Management dialog, and then select and operate the Autoload tab displayed on right-hand side, it is possible to change valid / invalid.



## 19 How to Exit the RSI-ECAT-Master

There are two kinds of end methods of RSI-ECAT-Master. One is that you finish a RSI-ECAT-Master process from 1INtime Explorer. The other is that you find the process handle of RSI-ECAT-Master to quit it by INtime API.



When INtime kernel is stopped without terminating RSI-ECAT-Master, the process of RSI-ECAT-Master does not finish, and there is a possibility that the situation (mainly freezing phenomenon) where a response does not return to Windows may occur.

RSI-ECAT-Master is cataloged as the name "RTECHDR" which is the process handle to the route process. This handle is searched with LookupRtHandle. Please terminate RSI-ECAT-Master by TerminateRtProcess using this process handle.

※ It takes about 5 seconds to end RSI-ECAT-Master.

## 20 Starting parameter

There is starting parameters which can be specified when starting RSI-ECAT-Master.

Default operation is performed when not specifying these parameters.

### 20.1 NIC selection parameters main line

`-if=[NIC driver name + Instance number]`

or

`-if=[0 ... 7]`

The NIC driver used by main line can be specified.



It is necessary to define two or more NIC drivers as the HPE2Ifs key which is in the [KPAMasterINtime] section of ecatmkpa.ini beforehand.

The NIC driver defined as the HPE2Ifs key can be specified as a parameter.



When the NIC driver which is not defined as a HPE2Ifs key is specified as a parameter, an error occurs and starting of RSI-ECAT-Master is interrupted.

When specifying `-if=[0 ... 7]`, the first NIC driver defined as the HPE2Ifs key is regarded as "0". Then select which NIC driver defined "1 , 2...7".

Default operation:

When there is no specification of this parameter, the first NIC driver defined as the HPE2Ifs key is used.

**【Example】**Specify that using instance 0 of ie100m

`-if=ie100m0`

**【Example】**Specify that using NIC driver 0th defined in the key HPE2Ifs

`-if=0`

## 20.2 NIC selection parameters redundancy line

-rif=[NIC driver name + Instance number]

or

-rif=[0 ... 7]

The NIC driver used by redundancy line can be specified.

(※ In order to realize redundancy, RSI-ECAT-Master/RED is needed.)

NICdriver name + Instance number which specified by main line cannot be specified.



**When you use the same kind of multiple NICs between main line and a redundancy line, please be careful not to overlap the instance number.**



It is necessary to define two or more NIC drivers as the HPE2Ifs key which is in the [KPAMasterINtime] section of ecatmkpa.ini beforehand

The NIC driver defined as the HPE2Ifs key can be specified as a parameter.



**When the NIC driver which is not defined as a HPE2Ifs key is specified as a parameter, an error occurs and starting of RSI-ECAT-Master is interrupted.**

When specifying -if=[0 ... 7], the first NIC driver defined as the HPE2Ifs key is regarded as “0”.  
Then select which NIC driver defined “1 , 2...7”.

Default operation:

You cannot do redundancy if this parameter is not specified.

**【Example】**Specify that using instance 1 of ie100m

-rif=ie100m1

**【Example】**Specify that using NIC driver 1st defined in the key HPE2Ifs

-if=1

### 20.3 Cyclic time specification parameter

`-cycle=[500 ...]`

You can specify the timing in which RSI-ECAT-Master performs cyclic process as micro second.

This value is based on the INtime kernel tick value.

So it is necessary to specify as the unit of the INtime kernel tick value.

Default operation:

When there is no specification of this parameter, it is set as the INtime kernel tick value.

**【Example】**INtime Kernel tick value :500  $\mu$  sec

Cyclic time specification:1000  $\mu$  sec

`-cycle=1000`

### 20.4 Thread base priority change parameter

`-prio=[10 ... 230]`

You can change the base priority of the thread which takes charge of sub process among the threads which RSI-ECAT-Master creates.

The thread which takes charge of sub process is adjusted to the lower priority on the basis of the specified value.

Default operation:

When there is no specification of this parameter, it is set as 130.

**【Example】**Base priority:Set 150

`-prio=150`

### 20.5 Starting delay parameter

`-delay=[1 ... 180000]`

Specify the time for standby before the initializing process after starting of RSI-ECAT-Master  
(Unit: Millisecond)

Default operation:

It does not wait when there is no specification of this parameter.

**【Example】**Specify wait 30 seconds

`-delay=30000`

## 20.6 Output parameters when error occurs

-eouth

Specify whether the output of the digital signal which the slave itself is outputting stops or keep on outputting as it is.

Now, the slave which correspond to this setting are as follows.

- (1) OMRON GX-OD1611
- (2) OMRON GX-OD3218

Default operation:

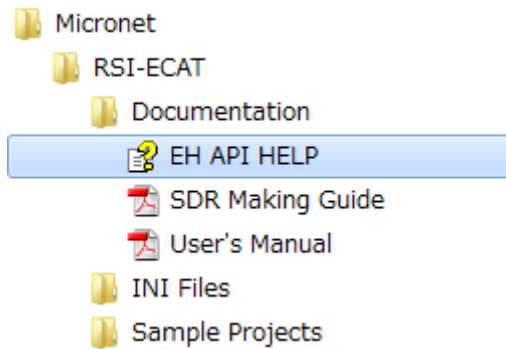
When there is no specification of this parameter, the output HOLD status is applied when errors has occurred.

## 21 API

RSI-ECAT-Master has two kinds of APIs.

Please refer to another file : EH API Help as for the details of API.

[Start menu] -> [All Programs] -> [Micronet] -> [RSI-ECAT] -> [Documentation] -> [EH API Help]



### 21.1 EH API

EH API is the API to use from the user applications.

There are APIs such as the API to access each category memory, the API to control the state machine of EtherCAT and so on.

The following composition is required in order to use EH API.

No	Additional composition items	Additional composition contents
1	Header file definition	#include <ehapi.h>
2	Additional header file path	\$(RSIECAT)¥include
3	Library file	EhApi.lib
4	Additional library path	\$(RSIECAT)¥lib

### 21.2 Slave API

Slave API is API used by SDR (Slave Driver Real-time shared library).

There are registration/acquisition of category memory, and registration/acquisition of PDI memory as main functions.

The following composition is required in order to use Slave API.

(※Since No.2,3 and 4 are the same definition as EH API, so are not required in case of “defined”.)

No	Additional composition items	Additional composition contents
1	Header file definition	#include <slvapi.h>
2	Additional header file path	\$(RSIECAT)¥include
3	Library file	EhApi.lib

4	Additional library path	\$(RSIECAT)¥lib
---	-------------------------	-----------------

## 22 File composition

The installation file compositions are as follows.

Installed in the following paths.

(Installation folder)\Micronet\RSI-ECAT

Each of following folder is installed.

Folder names and its contents are described.

bin folder

Folder containing the binary executable RSI-ECAT-Master.

bin\NodeA folder

Folder containing the definition file referred when performing RSI-ECAT-Master on NodeA.

When changing settings, you edit the ecatkpa.ini file in this folder.

(※ This setting is applied only when performing in NodeA.)

doc folder

Folder containing the documents of RSI-ECAT-Master.

EH-API HELP and this User's Manual exist.

include folder

Folder containing the header files to be included when you create the SDR or you use the EH-API.

lib folder

Folder containing the library file to be linked when you create the SDR or you use the EH-API.

sample folder

Folder containing the sample programs of RSI-ECAT-Master.

## 23 The concept of a node

RSI-ECAT-Master can start one RSI-ECAT-Master on one INtime kernel.

Therefore you can start one RSI-ECAT-Master on each INtime kernel in the environment where multiple INtime kernels can start. (That is to say, you can start multiple RSI-ECAT-Masters in such a PC.) In case of such environments, users who use EH-API knows INtime node and they can distinguish multiple RSI-ECAT-Masters by using the node names.

RSI-ECAT-Master itself knows by the node name folder(s) in the current folder of RSI-ECAT-Master installation folder (\RSI-ECAT\bin\).

(※ In case of the NodeA, see the file of \RSI-ECAT\bin\NodeA)

When using other nodes, you create the folder whose name is the same as the node name in the current of a RSI-ECAT-Master installation folder (\RSI-ECAT\bin\ ) as well as above “NodeA” folder, and then arrange the files to use.

**【Example】**When RSI-ECAT-Masters is started in NodeA, NodeB, and NodeC

< Set NodeB >

Create \RSI-ECAT\bin\NodeB folder

Assign files NodeB (RtEcHdr.xml) on \RSI-ECAT\bin\NodeB folder.

Assign files NodeB (ecatmkpa.ini) on \RSI-ECAT\bin\NodeB folder.

Create Node B by INtime Configuration

Pass NIC used by NodeB to to NodeB of INtime by INtime Configuration.

Add RtEcHdr.rta to Autoload of NodeB and make void by INtime Configuration.

Set INtime memory assignment for NodeB by INtime Configuration.

< Set NodeC >

Create \RSI-ECAT\bin\NodeC folder

Assign files NodeC (RtEcHdr.xml) on \RSI-ECAT\bin\NodeC folder.

Assign files NodeC (ecatmkpa.ini) on \RSI-ECAT\bin\NodeC folder.

Create Node C by INtime Configuration

Pass NIC used by NodeC to to NodeC of INtime by INtime Configuration.

Add RtEcHdr.rta to Autoload of NodeC and make void by INtime Configuration.

Set INtime memory assignment for NodeC by INtime Configuration.



### Cautions

When starting two or more RSI-ECAT-Masters, 1 RSI-ECAT-Master license is needed per 1 RSI-ECAT-Master. For details, please refer to [About license].

## 24 The check of a starting state

The way RSI-ECAT-Master checks a starting state are as follows.

### 24.1 Call the API

There is an API "EhGetEhNodeStatus()" which checks whether RSI-ECAT-Master has started or not.

You can check the starting state by using this API.

Moreover, it can also block returning from a function until RSI-ECAT-Master starts.

Please refer to another file:EH API Help for EhGetEhNodeStatus().

### 24.2 Check by INtime Explorer

RSI-ECAT-Master is cataloged as the name "RTECHDR" which is the process handle to the route process.

If the thread that "R? END\_INIT" is cataloged, it means RSI-ECAT-Master has been initialized and RSI-ECAT-Master is running.

## 25 Master default configuration

The Master default configurations are as follows.

No	Item	Default value	Change
1	CycleTime	Twice of an INtime kernel tick value	Yes
2	SubCycleTime	4 times of a cycle time value	No
3	AutoRecoveryTime	200 times of a cycle time value	No
4	NIC	ecatmkpa.ini HPE21fs key 0th definition	Yes
5	Redundancy	Not used	Yes

**【Example】**INtime kernel tick value: 500us

CycleTime = 1ms

SubCycleTime = 4ms

AutoRecoveryTime = 200ms

## 26 ecاتمkpa.ini setting item

It is explanation about each item of the ecاتمkpa.ini file edited in order to set up NIC.

```
[KPAMaster]
;Master task base priority
BaseTaskPriority=80

[KPAMasterINtime]
;Space separated list of available HPE2 devices
; Possible values :
;   ie1gN    - Intel 1G device (N represents the unit number,
;               e.g. "ie1g0" for the first device assigned to INtime kernel)
;   ie100mN  - Intel 10/100 device
;   rtl1gN   - RealTek 1G device
;   rtl100mN - RealTek 10/100 device
HPE2Ifs=ie1g0 ie100m0 rtl1g0 rtl100m0
```

There are two items which can be set up.



※ Do not change sections and key items other than the item currently mentioned here.

### (1) BaseTaskPriority

It is a setting of the base priority for threads of an EtherCAT Master protocol stack.

It has been set to the default 80. Thread of higher priority are not generated than this value.

Please set only to make a priority lower than 80.

※ There is a possibility that stable communication cannot be done when you make the priority lower.

### (2) HPE2Ifs

It is a definition of the NIC drivers which an EtherCAT Master protocol stack uses. This NIC driver definition consists of "NIC driver name + instance numbers."

To this key, you can define to a maximum of eight items of NIC driver by separating by a space.

When not using a parameter, an EtherCAT Master protocol stack uses the first NIC driver registered into this key.



For information on parameters, refer to the [ [NIC selection parameters main line](#) ] and [ [NIC selection parameters redundancyline](#) ] .

## 27 Sample Code

The following sample codes describe the process such as initialization, device detection, State change, I/O control.

```

1  /*****
2  * Description      : RSI-ECAT-Master : DO Program sample
3  *****/
4  #include <stdio.h>
5  #include <rt.h>
6  #include <EhApi.h>    // EH API Header file
7
8  /*****
9  * main process
10 *****/
11 void main(int argc, char* argv[])
12 {
13     WORD        woStatus;
14     WORD        woReqState = 0;
15     WORD        woNowState = 0;
16     EHHANDLE     hAPI;
17     SLAVE_DETAIL stSlave;
18
19     // Check RSI-ECAT-Master Start status (Call RSI-ECAT API)
20     woStatus = EhGetEhNodeStatus("NodeA", 100);
21     if( 0 != woStatus ) {
22         printf("EhGetEhNodeStatus() Status = %04xH\r\n", woStatus);
23         printf("start check failed\r\n");
24         return ;
25     }
26
27     // RSI-ECAT API Dialog (Conversation) handle OPEN (Call RSI-ECAT API)
28     hAPI = EhOpen("NodeA", &woStatus);
29     if( 0 != woStatus ) {
30         printf("EH Handle Open failed\r\n");
31         return ;
32     }
33
34     // Request Master state change (Call RSI-ECAT API)
35     // MST_OPERATIONAL : Transition to operational
36     woStatus = EhRqState(hAPI, MST_OPERATIONAL);
37     if( 0 != woStatus ) {
38         printf("EhRqState() failed. Status = %04xH\r\n", woStatus);
39         return ;
40     }
41
42     // Checking whether a transition to operational
43
44     while(1) {
45         // Gets the current state of the Master state (Call RSI-ECAT API)
46         woStatus = EhGetState(hAPI, &woReqState, &woNowState);
47
48         // Check whether the completion of the transition to operational
49         if( woNowState == MST_OPERATIONAL ) {
50             break; // Break because of the completion transition
51         }
52
53         // Sleep wait for state change
54         RtSleep(500);
55     }

```

Checked whether RSI-ECAT-Master has started. Here, maximum of 100 msec standby until it'll start. If cannot start, end. It is also possible to stand by until RSI-ECAT-Master starts the second argument by using WAIT\_FOREVER.

Open an API dialog (=conversation) handle required for an EH API call. Thereby, start to connect with a RSI-ECAT-Master process.

Request a state change of the Master. Requirement of transition from the initial state to the operational state.

Check whether the State of Master has been changed. While to loop is under execution until it is changed. Although the State change of Master come under the influence of the number of slaves or the connection environment, it takes about ten seconds to about dozens of seconds.

Issuing a sleep waiting for the completion of the Master state change

```

56 // Access process to a slave from here
57
58 // Search slave
59 stSlave.dwSize = sizeof(stSlave);
60 stSlave.dwVendorID = 131;
61 stSlave.dwProductCode = 50;
62 stSlave.dwInstance = 0;
63
64 woStatus = EhFindSlave(hAPI, &stSlave);
65 if( 0 != woStatus ) {
66     printf("EhFindSlave() failed. Status = %04xH\n", woStatus);
67     return ;
68 }
69
70 // Process of operation
71 while(1) {
72     // Waiting for cyclic process
73     woStatus = EhWaitForCyclic(hAPI, WAIT_FOREVER);
74     if( 0 != woStatus ) {
75         break; // End because cyclic process is interrupted
76     }
77
78     // Conversion request ON / OFF per each loop
79     if( 0 != EhReadbackByte(hAPI, stSlave.dwViosOutBaseOffset, &woStatus) ) {
80         EhWriteByte(hAPI, stSlave.dwViosOutBaseOffset, 0x00);
81     } else {
82         EhWriteByte(hAPI, stSlave.dwViosOutBaseOffset, 0xFF);
83     }
84 }
85 printf("program end.");
86
87 }

```

In searching for a slave,  
 ①VendorID, ②ProductCode ③instance number are required.  
 Here, require to detect the slave with the slave of "VendorID = 131 and ProductCode = 50 and instance no = 0 (First Detected)". When searching two or more same slaves, slave information returns in the order in which the instance number is found by carrying out an increment.

Possible to block until cyclic process occurs by setting the time(\*) as WAIT\_FOREVER.  
 (the time(\*) means one of waiting for API which returns response every cyclic process of EtherCAT Master)

The input and output to PI are performed by API which performs VIOS access.  
 0 and 0xFF are reversed and outputted based on the last output value.

In order to build this sample code, the following project settings are needed.

- ① Add \$(RSIECAT)\include as an additional path of the header file.
- ② Add \$(RSIECAT)\lib as the path of the additional library.
- ③ Add EhApi.lib as additional library.

## 28 Sample Program

Sample programs are EH API test tool which uses all EH APIs.

### 28.1 How to run the sample program

In order to run a sample program, it is necessary to start an INtime kernel beforehand.

The method of starting an INtime kernel is as follows.

It right-clicks on the INtime icon of a task tray, and an INtime kernel starts by click [NodeA]–[Start NodeA].

Note that “NodeA” is a default name of an INtime kernel.

To perform various kinds of EH APIs, it is necessary to start RSI-ECAT-Master (RtEcHdr.rta).

Please refer to [ [How to start the RSI-ECAT-Master](#) ] for the starting method of RSI-ECAT-Master.

Please run the sample program RTA from the following path.

#### For Windows XP:

(My Documents)\Documents\Micronet\RSI-ECAT\Projects\EhApiSample\EhApiSample.rta

#### For Windows 7:

C:\Users\ (Account name) \Documents\Micronet\RSI-ECAT\Projects\EhApiSample\EhApiSample.rta

## 28.2 Checking of Operations of Sample Program

One console window will be displayed by executing sample programs.

The API groups to which the number was assigned will be displayed on the console window.

Each API can be performed by pressing the Enter key after inputting a half-width numerical value.

```

RT I/O Console 0x1F68 - NodeA
EhApiSample started
EhApiSample finished initialization
I$ RSI-ECAT EH API Sample Start.

ID : FUNCTION NAME      | ID : FUNCTION NAME
-----
0: Menu list            | 99: Exit
1: EhGetEhNodeStatus    | 2: EhOpen
3: EhClose              | 4: EhRqState
5: EhGetState           | 6: EhGetSlaveCount
7: EhGetOnlineSlaveCount | 13: EhGetEventFilter
14: EhSetEventFilter    | 15: EhWaitForEvent
16: EhReadOD            | 17: EhWriteOD
18: EhDiGetChNums       | 19: EhDiRead
20: EhDiBlock           | 21: EhDoGetChNums
22: EhDoRead            | 23: EhDoWrite
24: EhDoBlock           | 25: EhAiGetChNums
26: EhAiRead            | 27: EhAiBlock
28: EhAiGetRange        | 29: EhAiSetRange
30: EhAoGetChNums       | 31: EhAoRead
32: EhAoWrite           | 33: EhAoBlock
34: EhAoGetRange        | 35: EhAoSetRange
36: EhWaitForCyclic     | 37: EhFindSlave
38: EhFindSlaveByAlias  | 39: EhGetSlaveStatus
40: EhGetSystemInfo     | 41: EhGetViosInOffset
42: EhGetViosOutOffset  | 43: EhReadODByAlias
44: EhWriteODByAlias    | 45: EhReadByte
46: EhReadWord          | 47: EhReadDword
48: EhWriteByte         | 49: EhWriteWord
50: EhWriteDword        | 51: EhReadbackByte
52: EhReadbackWord      | 53: EhReadbackDword
Set Menu ID >>
  
```

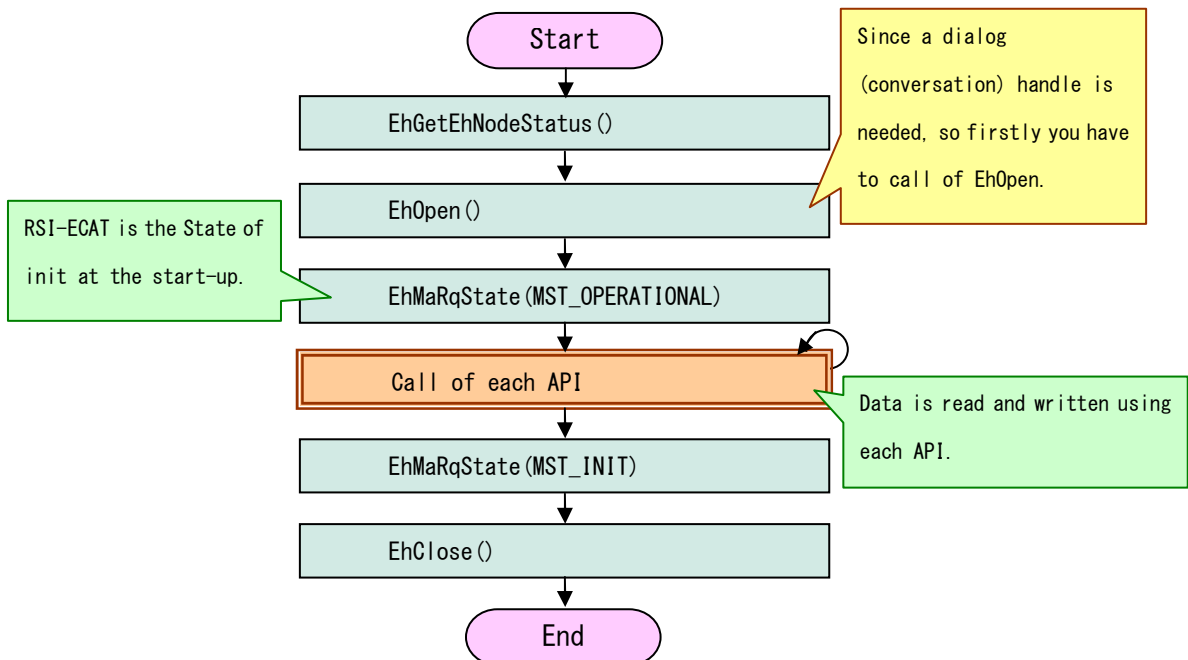
The method of the menu display and the method of quitting of a program are as follows.

No	Item	Command
1	Display menu	Input 0 and Enter Key
2	End of the program	Input 99 and Enter Key

Please refer to EH API Help as for each API.



The procedure of performing API is as follows.



### 28.3 Edit of sample program

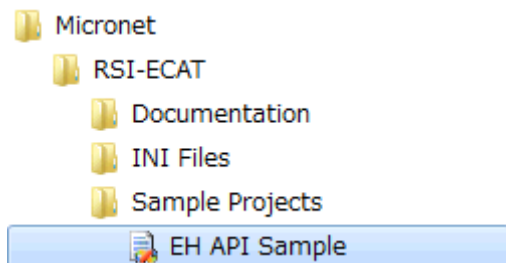
Sample programs are edited by "Visual Studio 2008".



※ INtime development environment is needed for the solution project for INtime applications. This solution cannot be opened in an environment without INtime development environment.

Sample programs can be edited from the following path.

[ Start ] -> [ All Programs ] -> [ Micronet ] -> [ RSI-ECAT ] -> [ Sample Projects ] -> [ EH API Sample ]



## 29 Troubleshooting

When error message is displayed from RSI-ECAT-Master, please check the item of an error message table.



### 29.1 The view of an error message table

Troubleshooting table is written as follows.

Error item number	Displayed error message
1	Error! Slave's attach failed. ST[xxxxH] Please check a slave's connect.
The communication error has occurred. Cannot access to EtherCAT slave.	
①	There is a possibility that EtherCAT slave is unconnected. →Connect with the EtherCAT slave.
②	There is a possibility that Ethernet to connect is wrong. →When there is two or more connectors, please connect with the connector which must be connected originally.
③	There is a possibility that the EtherCAT slave of the error holding circuit ON exists. →Please turn OFF on an error holding circuit, and turn off and turn on the power supply of a slave, then start again.
④	There is a possibility that the connection is shut down and ended during the last connection. →Please turn off the power supply of a slave, turn on in several seconds, and start again.
⑤	The power supply of an EtherCAT slave may be OFF. → Please turn ON the power supply of an EtherCAT slave.

Item number of Condition assumption

Contents of condition assumption

Trouble action method

## 29.2 Error Message Table

Any errors can not be recovered from that state. You will always need to restart the RSI-ECAT-Master.

1	<p>Error! Slave's attach failed. ST[xxxxH]  Please check a slave's connect.</p>
The communication error has occurred. Cannot access to EtherCAT slave.	
①	<p>There is a possibility that EtherCAT slave is unconnected.  →Connect with the EtherCAT slave.</p>
②	<p>There is a possibility that Ethernet to connect is wrong.  →When there is two or more connectors, please connect with the connector which must be connected originally.</p>
③	<p>There is a possibility that the EtherCAT slave of the error holding circuit ON exists.  →Please turn OFF on an error holding circuit, and turn off and turn on the power supply of a slave, then start again.</p>
④	<p>There is a possibility that the connection is shut down and ended during the last connection.  →Please turn off the power supply of a slave, turn on in several seconds, and start again.</p>
⑤	<p>The power supply of an EtherCAT slave may be OFF.  → Please turn ON the power supply of an EtherCAT slave.</p>
⑥	<p>There is a possibility that initialization of NIC and an EtherCAT slave failed.  →Restart Windows, and try to start the RSI-ECAT-Master OFF → ON after the power of the slave.</p>
2	<p>Error! XML file is cyclic command not supported.  Please select [Use LRW Command] in Studio EtherCAT.</p>
The definition of an ENI (EtherCAT Network Information) file is not a LRW command method.	
①	<p>There is a possibility that it is not set as a LRW command method when you create ENI file.  →Remake an ENI file by RSI-ECAT-Studio (option).</p>
3	<p>Error! Alias duplicate.  Please check a slave's alias.</p>
The slave which has the same alias number in the connected slaves exists.	
①	<p>There is a possibility of duplication as the alias number or hard ID for each slave.  →Please check the alias number by RSI-ECAT-Studio (option) and correct not to overlap it.  →Please check the alias number currently assigned to the slave and correct not to overlap it.</p>

4	Error! ecاتمkpa.rsl initialization failed. ST[xxxxxxxxH]
ecatmkpa.rsl required for RSI-ECAT-Master cannot be loaded.	
①	ecatmkpa.rsl may not exist. →Please check existence of (RSI-ECAT-Master installation path)\bin\ecatmkpa.rsl If it exists, please reinstall.

5	Error! AllocateMemory and/or CatalogObject failed. RET[xx] ST[xxxxH]
Initialization of RSI-ECAT-Master failed.	
①	There is a possibility of being short of INtime memory. →Please check whether the INtime memory is set as 64 MB or more for RSI-ECAT-Master. In addition, when the INtime process which is operating exists, please set up the value which is also considered for the amounts for the memory of other processes.
②	There is a possibility of being short of the numbers of INtime object directories. →Please open INtime Explorer and check the numbers of object directories of a PRGS_RTECHDR process. There is the notation [ Object directory has xxx entries, xxx in use ]. When all the values of xxx are being used, please increase the number of object directories from Node Management of INtime Configuration. RSI-ECAT-Master consumes the number of object directories 100.

6	Error! Cannot open RtEcHdr.XML file. errno[xx] Please check an XML path and file exist. XML path[.....]
Opening RtEcHdr.XML failed.	
①	RtEcHdr.XML file may not exist. →Please check whether RtEcHdr.XML exists on the path shown in displayed XML path. Arrange RtEcHdr.XML when it does not exist. When it does not exist, check file attributes such as a hidden attribute, and eliminate the attribute. When it exists and no file attributes such as a hidden attribute, there is a possibility of locking the file by another application. Please cancel a file lock.

7	<p>Error! Cannot open ecاتمkpa.ini file. errno[xx]</p> <p>Please check file exist.</p> <p>File[.....]</p>
	<p>ecatmkpa.ini open file failed.</p>
①	<p>ecatmkpa.ini file may not exist.</p> <p>→Please check whether ecاتمkpa.ini exists on the path shown in displayed File path.</p> <p>Arrange ecاتمkpa.ini when it does not exist.</p> <p>When it does not exist, check file attributes such as a hidden attribute, and eliminate the attribute.</p> <p>When it exists and no file attributes such as a hidden attribute, there is a possibility of locking the file by another application. Please cancel a file lock.</p>
8	<p>Error! Ethernet adapter attach failed.</p> <p>Please check an ethernet adapter. ST[xx]</p> <p>selected ethernet 1th driver : [.....]</p> <p>selected ethernet 2th driver : [.....]</p>
	<p>Starting NIC driver failed.</p>
①	<p>NIC driver is not found.</p> <p>→Add a definition to the <b>HPE21fs</b> key of ecاتمkpa.ini.</p>
②	<p>Specification of the NIC driver is wrong.</p> <p>→Correct the item defined as the <b>HPE21fs</b> key of the ecاتمkpa.ini.</p>
③	<p>The value of the starting parameter (-if) is wrong.</p> <p>→ The numerical value which can be specified as "-if" is from 0 to 7.</p> <p>→ The character string which can be specified as "-if" is only the NIC driver defined as the ecاتمkpa.ini.</p>
④	<p>The value of the starting parameter (-rif) is wrong.</p> <p>→ The numerical value which can be specified as "-rif" is from 0 to 7.</p> <p>→ The character string which can be specified as "-rif" is only the NIC driver defined as the ecاتمkpa.ini.</p>

9	Error! Wrong startup parameter. Please check startup parameter.
Startup parameter specification is wrong.	
①	Specifying startup parameters mistakes or the range of values is over. →Please check and correct the startup parameter.

10	Error! Cycle-Time wrong. Cycle-Time[xxx usec] INtime-Kernel-Ticks[xxx usec] Please check a parameter Cycle-Time(-cycle=xxx) value or INtime-Kernel-Ticks.
EtherCAT cyclic time is wrong.	
①	<p>The value of -cycle=xxx specified with the startup parameter cannot be performed. →Specify by the unit of an INtime kernel tick value.</p> <p>For example, when a kernel tick value is 500 microseconds, you can set a cyclic time only as 500 microseconds interval like 1000 microseconds, 1500 microseconds, 2000 microseconds ---.</p> <p>If a kernel tick value is 250microseconds, you can set a cyclic time as 250 microseconds interval. You can set at an interval of "250 microseconds" like 500 microseconds, 750 microseconds, 1000 microseconds ---.</p> <p>→Please change an INtime kernel tick value.</p> <p>When the INtime kernel tick value is larger than one of a cyclic time, you cannot realize even though it is minimum kernel tick value, either.</p> <p>In this case, it is necessary to set an INtime kernel tick value as a smaller value.</p> <p>For example, when a kernel tick value is 1000 microseconds and a cyclic time is 500 microseconds, Since 500 microseconds of a cyclic time cannot be realized, so it is necessary to set it as a realizable kernel tick value such as 500 microseconds, 250 microseconds, 100 microseconds etc.</p>

11	Error! RSI-ECAT License failed. errno[xx] Signature:xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx Register :xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
RSI-ECAT-Master license error.	
①	Dongle keys (USB, LPT, etc.) : The license of RSI-ECAT-Master may not be applied. → RSI-ECAT-Master can be used only with PC in which the dongle key (USB, LPT, etc.) is registered when you purchased RSI-ECAT-Master. It cannot be used in other PCs.
②	Activator: The license of RSI-ECAT-Master may not be applied. → RSI-ECAT-Master can be used only with PC in which the activator is registered when you purchased RSI-ECAT-Master. It cannot be used in other PCs.

## 30 About License

One license per one node is required for RSI-ECAT-Master.

e.g. If you'd like to start RSI-ECAT-Masters on four nodes, respectively, four RSI-ECAT-Master licenses are needed.

Guidance of the contact about a license:

■ Contact by telephone

TEL +81-299-90-1733 time 9:30~17:30 (From Monday to Friday except a public holiday)

■ Contact by FAX

FAX +81-299-92-8557 (24-hour correspondence)